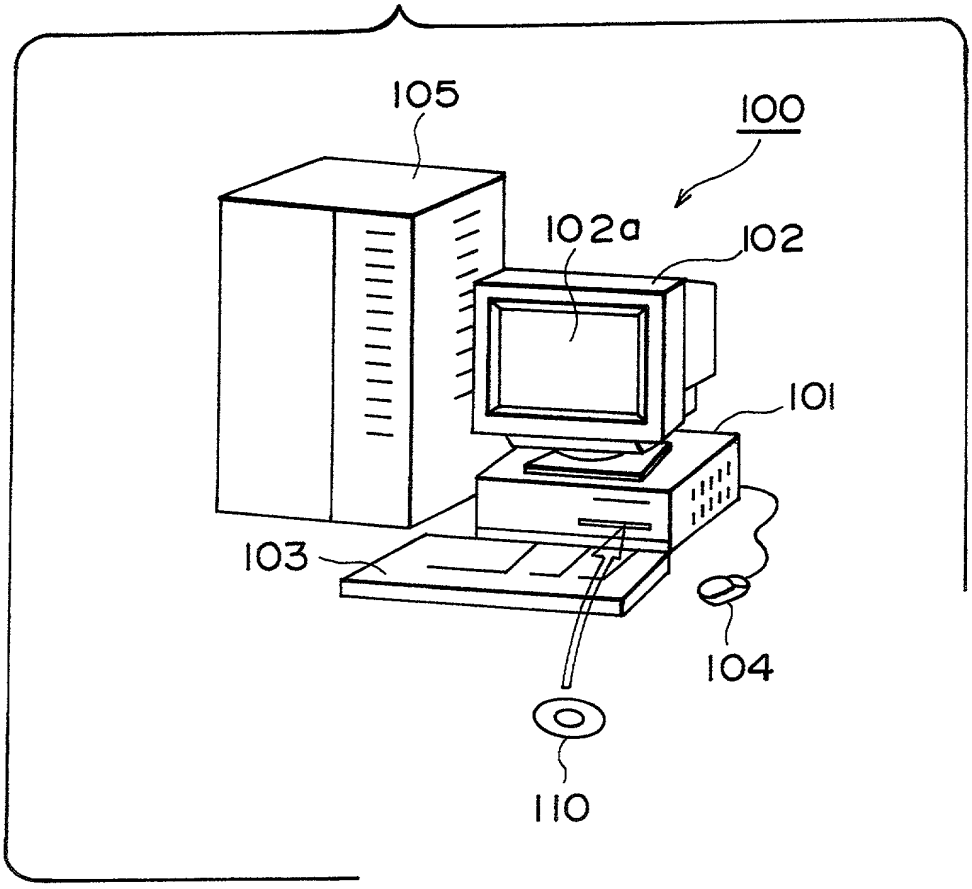
Fig. 1



EPO - DG 1 09.04.2003 09:37:56

Fig. 2

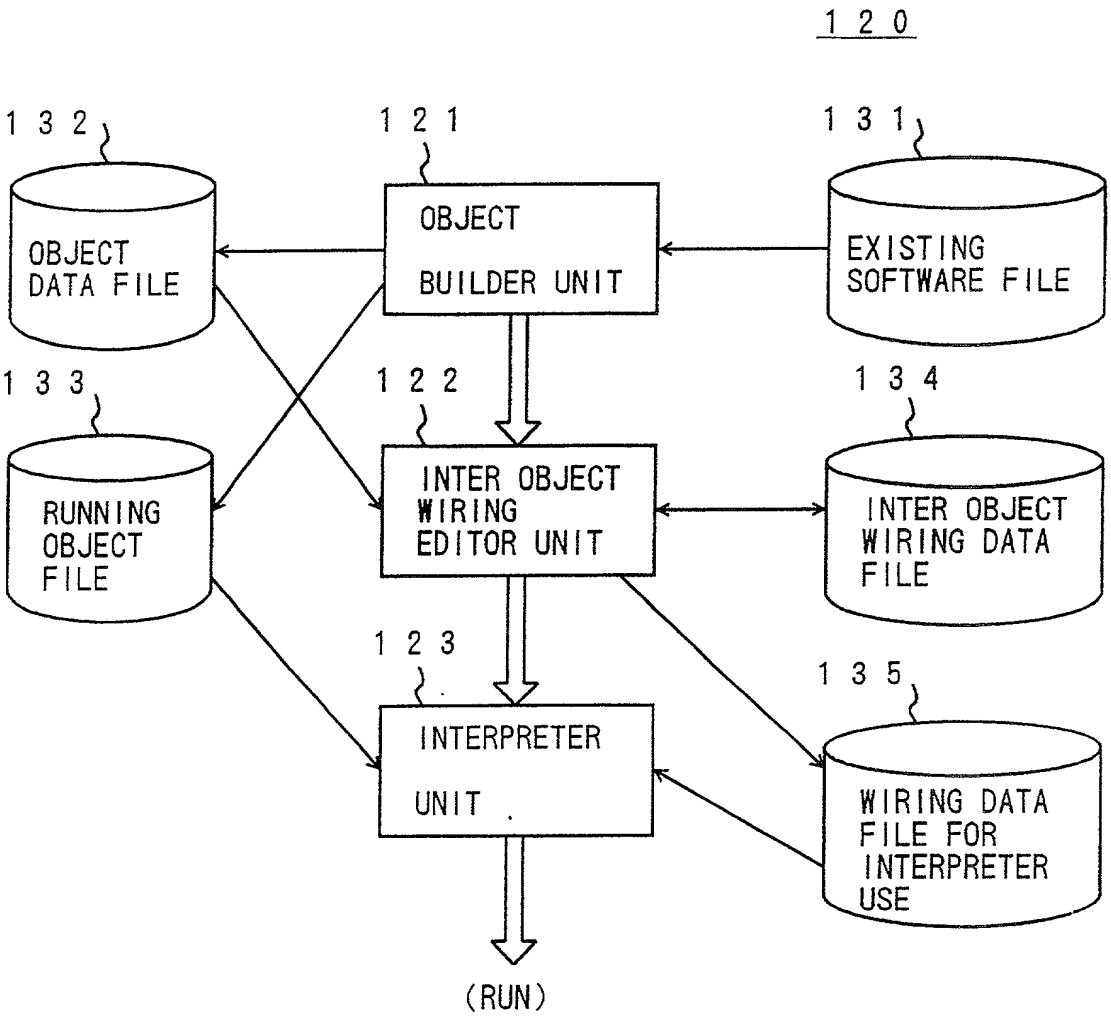

120
132
121
131
OBJECT DATA FILE
OBJECT BUILDER UNIT
EXISTING SOFTWARE FILE
133
122
134
RUNNING OBJECT FILE
INTER OBJECT WIRING EDITOR UNIT
INTER OBJECT WIRING DATA FILE
123
135
INTERPRETER UNIT
WIRING DATA FILE FOR INTERPRETER USE
(RUN)

Fig. 3

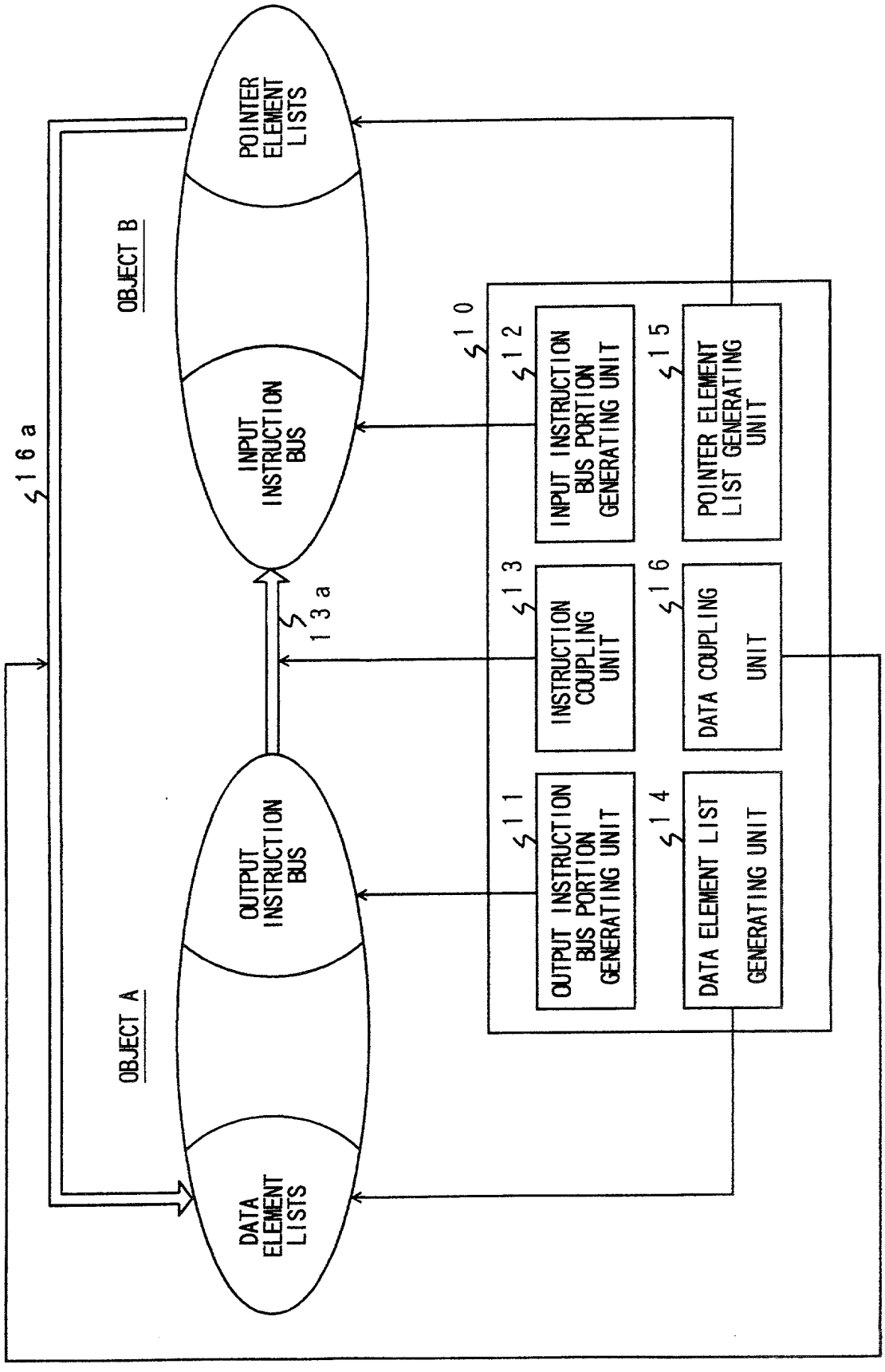

POINTER ELEMENT LISTS
INPUT INSTRUCTION BUS
OUTPUT INSTRUCTION BUS
DATA ELEMENT LISTS
OBJECT B
OBJECT A
16a
13a
10
12
INPUT INSTRUCTION BUS PORTION GENERATING UNIT
15
POINTER ELEMENT LIST GENERATING UNIT
13
INSTRUCTION COUPLING UNIT
16
DATA COUPLING UNIT
11
OUTPUT INSTRUCTION BUS PORTION GENERATING UNIT
14
DATA ELEMENT LIST GENERATING UNIT

Fig. 4

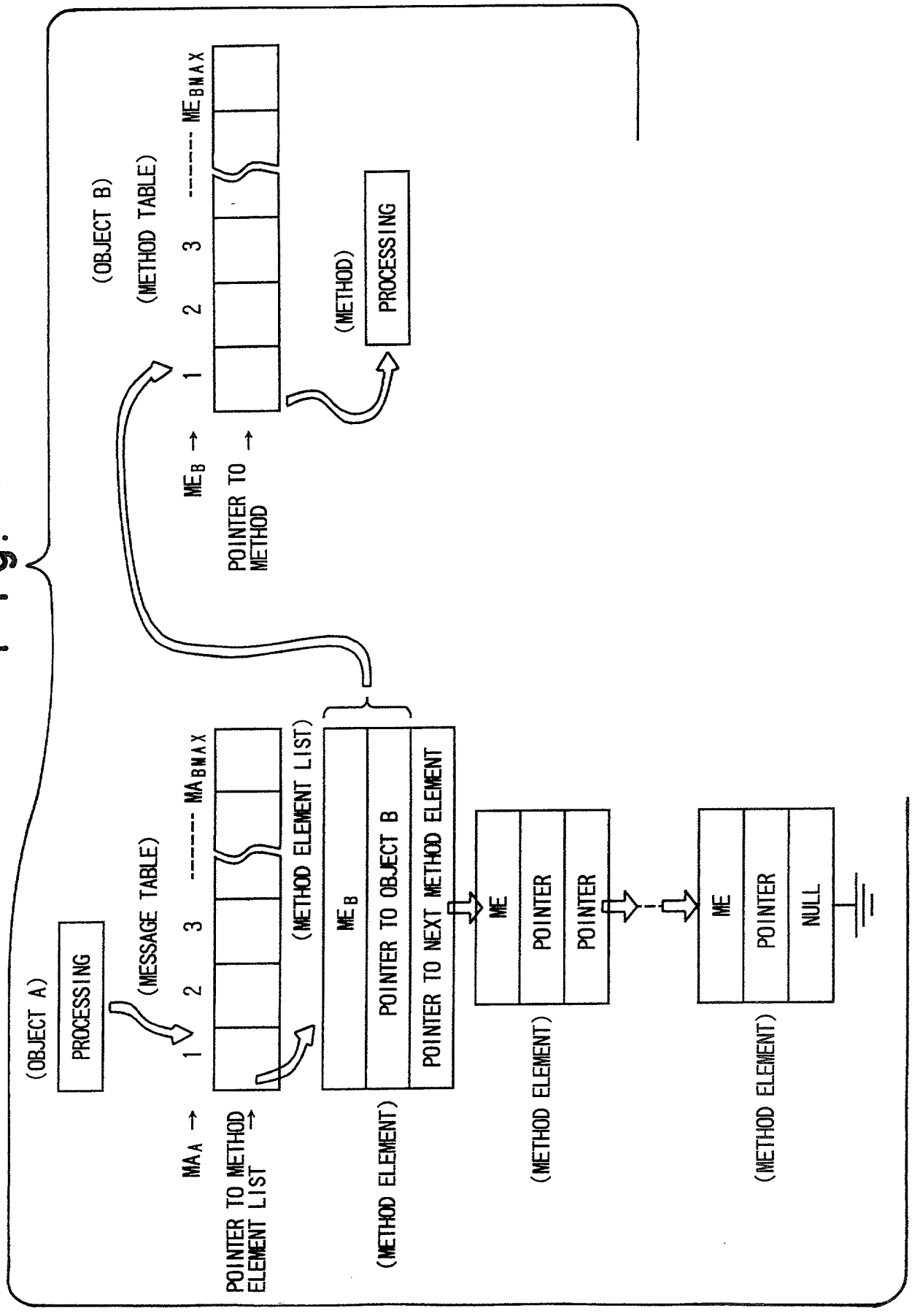

(OBJECT A)
PROCESSING
(MESSAGE TABLE)
1 2 3 ------ MA BMAX
MA A →
POINTER TO METHOD ELEMENT LIST
(METHOD ELEMENT LIST)
ME B
POINTER TO OBJECT B
POINTER TO NEXT METHOD ELEMENT
(METHOD ELEMENT)
ME
POINTER
POINTER
(METHOD ELEMENT)
ME
POINTER
NULL
(OBJECT B)
(METHOD TABLE)
1 2 3 ------ ME BMAX
ME B →
POINTER TO METHOD
(METHOD)
PROCESSING

Fig. 5

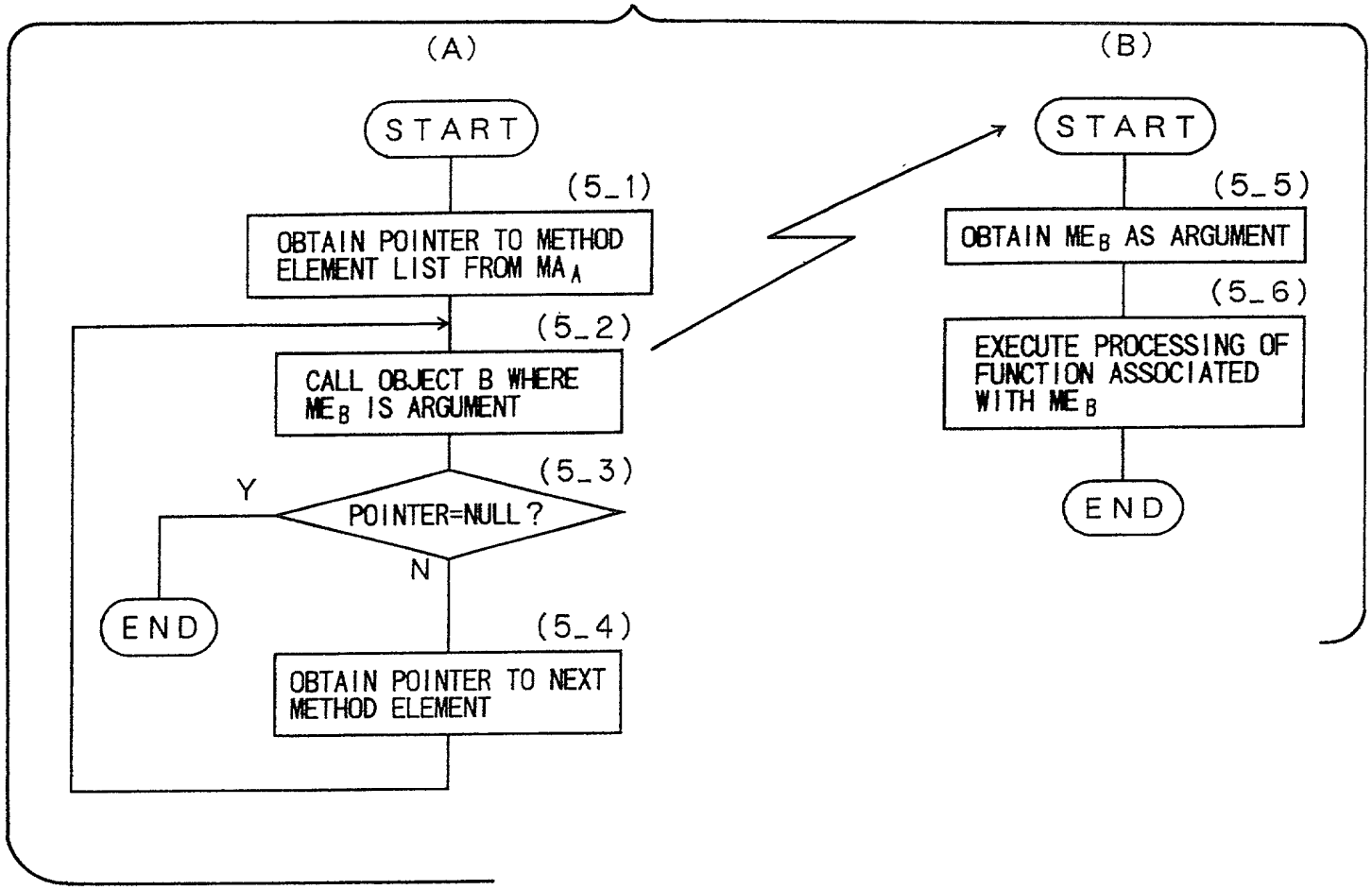

(A)
START
(5_1)
OBTAIN POINTER TO METHOD ELEMENT LIST FROM MA A
(5_2)
CALL OBJECT B WHERE ME B IS ARGUMENT
(5_3)
Y
POINTER=NULL?
N
END
(5_4)
OBTAIN POINTER TO NEXT METHOD ELEMENT
(B)
START
(5_5)
OBTAIN ME B AS ARGUMENT
(5_6)
EXECUTE PROCESSING OF FUNCTION ASSOCIATED WITH ME B
END

Fig. 6

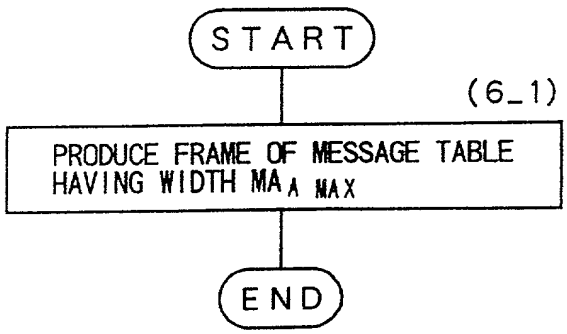

START
(6_1)
PRODUCE FRAME OF MESSAGE TABLE HAVING WIDTH MA A MAX
END

Fig. 7

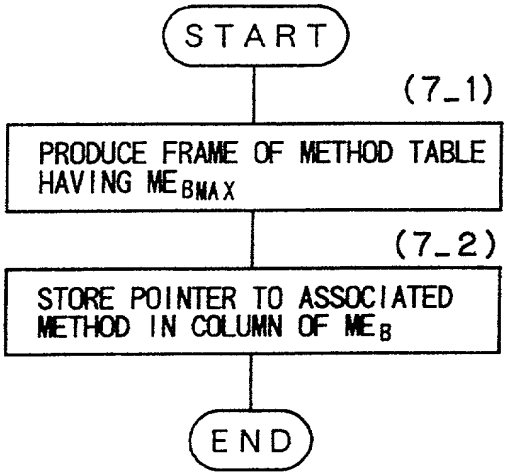

START
(7_1)
PRODUCE FRAME OF METHOD TABLE HAVING ME_{BMAX}
(7_2)
STORE POINTER TO ASSOCIATED METHOD IN COLUMN OF ME_B
END

Fig. 8

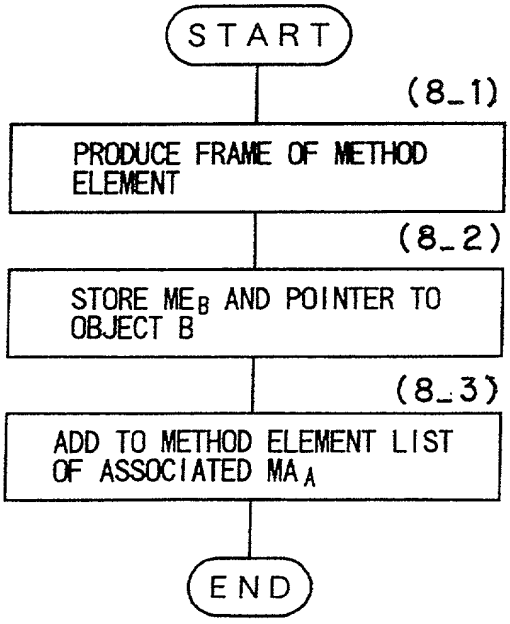

START
(8_1)
PRODUCE FRAME OF METHOD ELEMENT
(8_2)
STORE ME_B AND POINTER TO OBJECT B
(8_3)
ADD TO METHOD ELEMENT LIST OF ASSOCIATED MA_A
END

Fig. 9

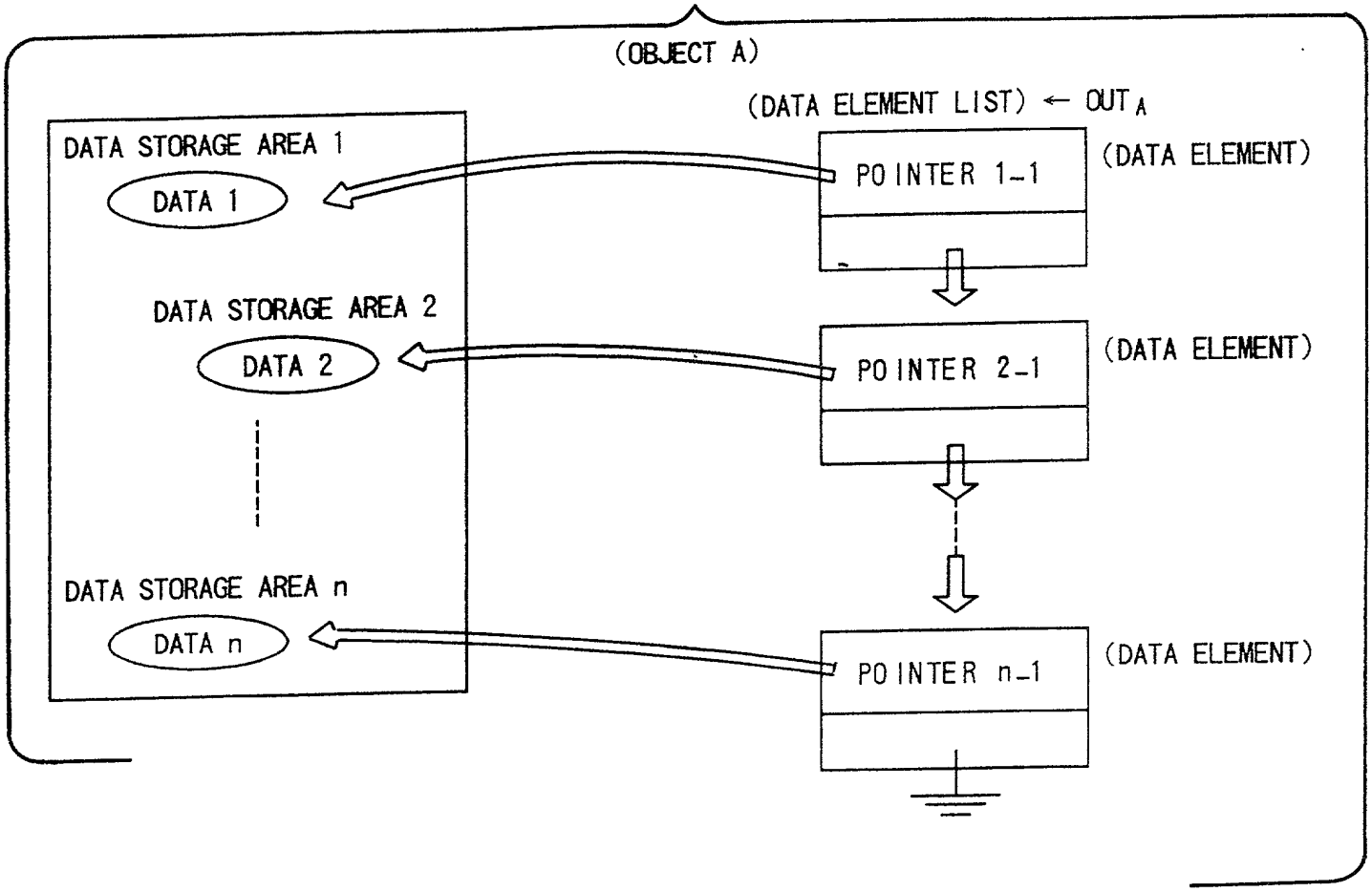

(OBJECT A)
(DATA ELEMENT LIST) ← OUT A
DATA STORAGE AREA 1
DATA 1
DATA STORAGE AREA 2
DATA 2
DATA STORAGE AREA n
DATA n
POINTER 1-1
POINTER 2-1
POINTER n-1
(DATA ELEMENT)
(DATA ELEMENT)
(DATA ELEMENT)

Fig. 10

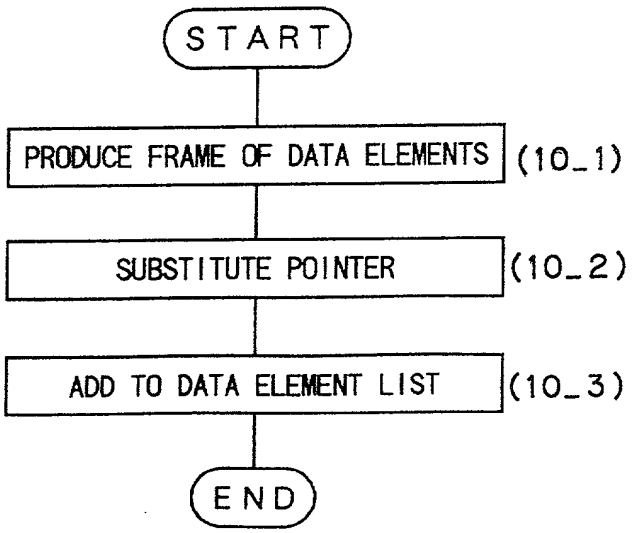

START
PRODUCE FRAME OF DATA ELEMENTS
(10_1)
SUBSTITUTE POINTER
(10_2)
ADD TO DATA ELEMENT LIST
(10_3)
END

Fig. 11

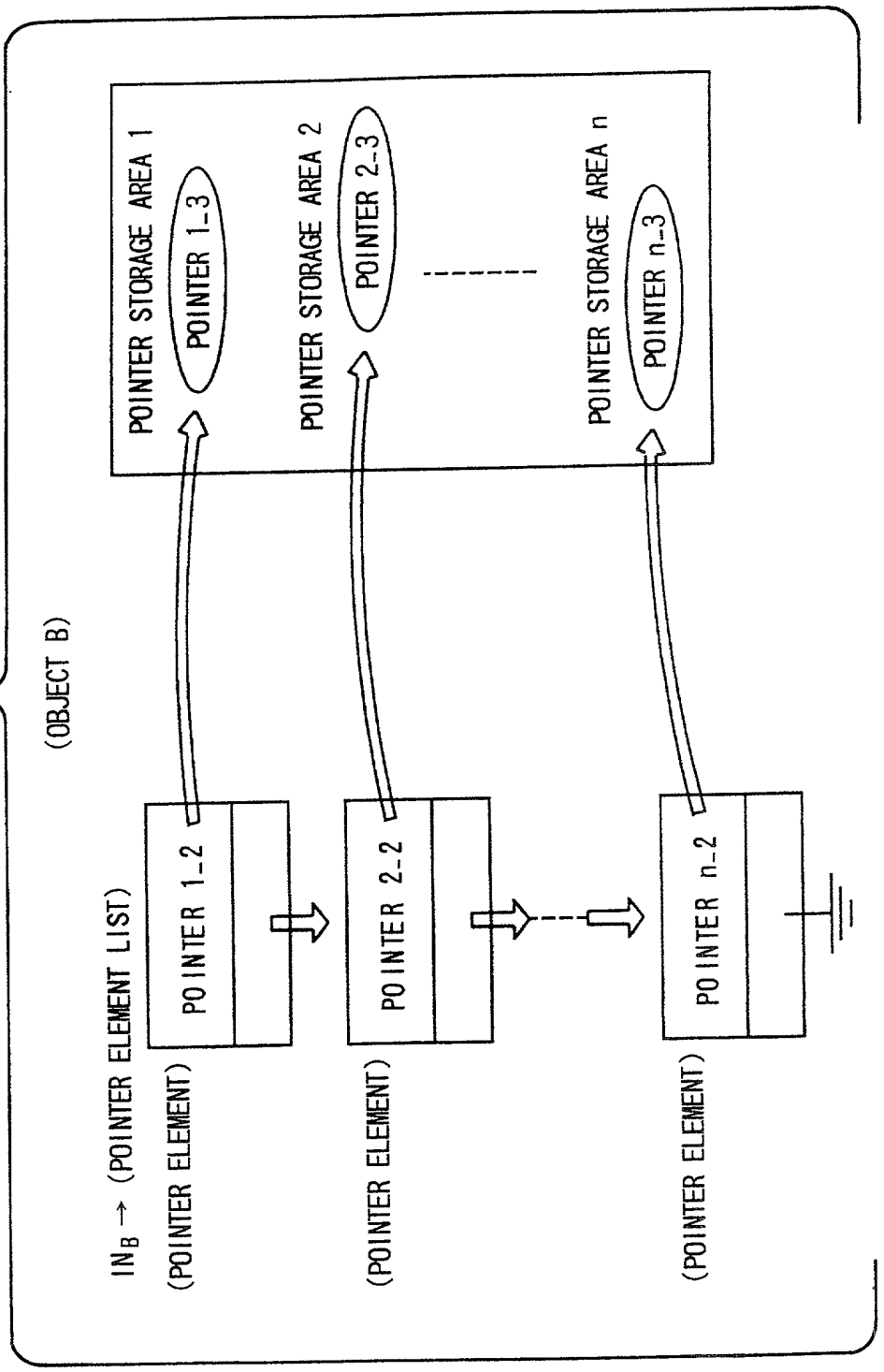

(OBJECT B)
POINTER STORAGE AREA 1
POINTER 1-3
POINTER STORAGE AREA 2
POINTER 2-3
POINTER STORAGE AREA n
POINTER n-3
IN_B → (POINTER ELEMENT LIST)
(POINTER ELEMENT)
POINTER 1-2
(POINTER ELEMENT)
POINTER 2-2
(POINTER ELEMENT)
POINTER n-2

Fig. 12

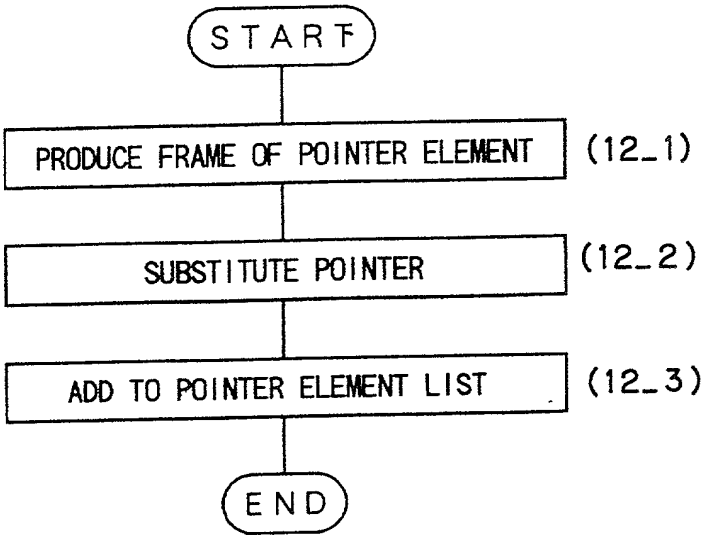

START
PRODUCE FRAME OF POINTER ELEMENT
(12_1)
SUBSTITUTE POINTER
(12_2)
ADD TO POINTER ELEMENT LIST
(12_3)
END

Fig. 13

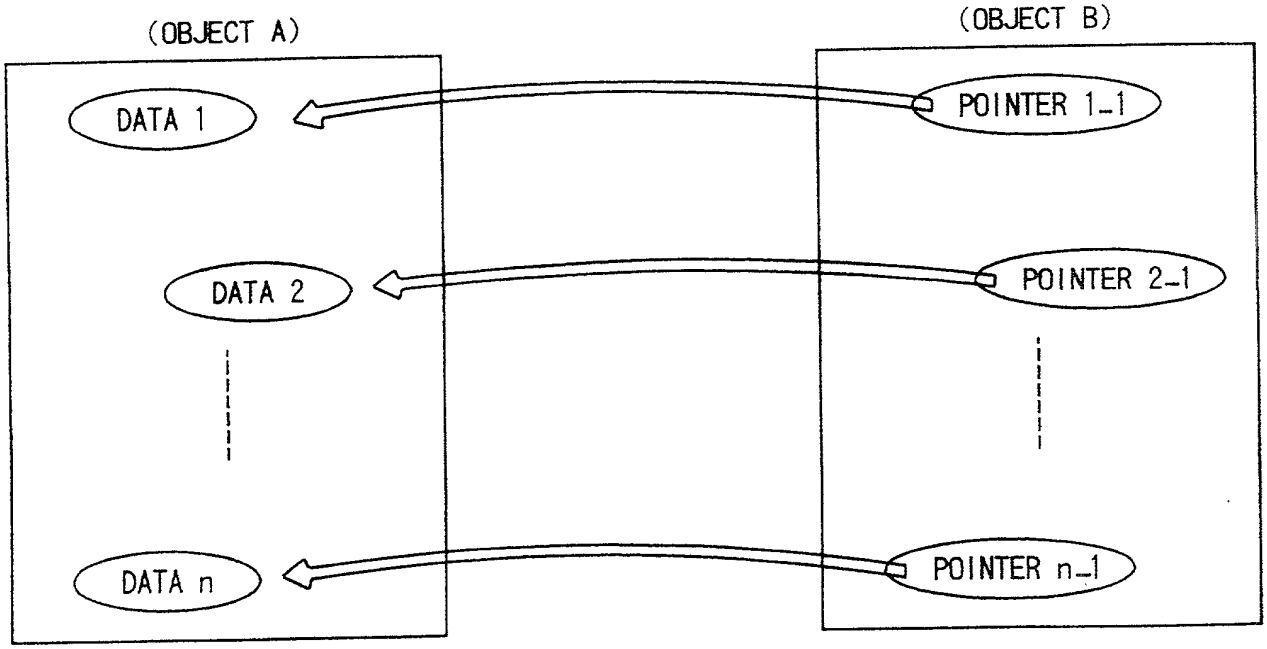

(OBJECT A)
(OBJECT B)
DATA 1
DATA 2
DATA n
POINTER 1_1
POINTER 2_1
POINTER n_1

Fig. 14

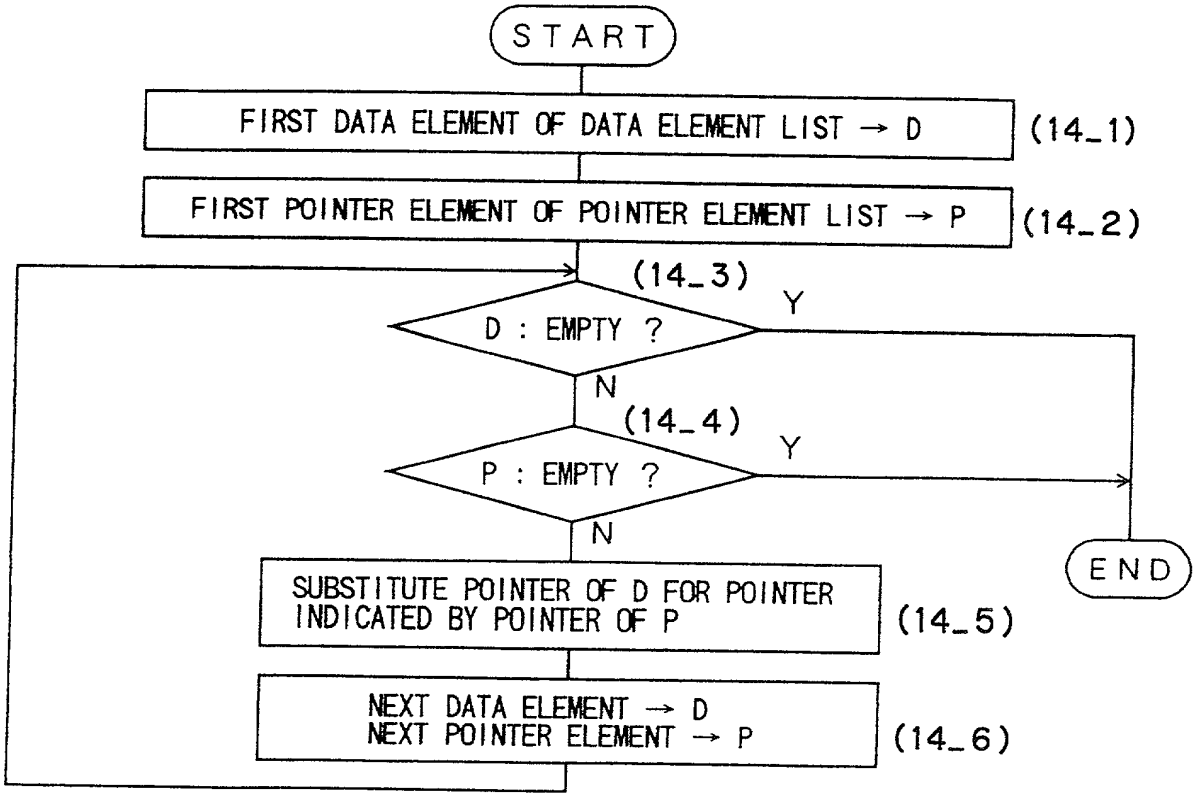

START
FIRST DATA ELEMENT OF DATA ELEMENT LIST → D
(14_1)
FIRST POINTER ELEMENT OF POINTER ELEMENT LIST → P
(14_2)
(14_3)
D : EMPTY ?
Y
N
(14_4)
P : EMPTY ?
Y
N
END
SUBSTITUTE POINTER OF D FOR POINTER INDICATED BY POINTER OF P
(14_5)
NEXT DATA ELEMENT → D
NEXT POINTER ELEMENT → P
(14_6)

Fig. 15

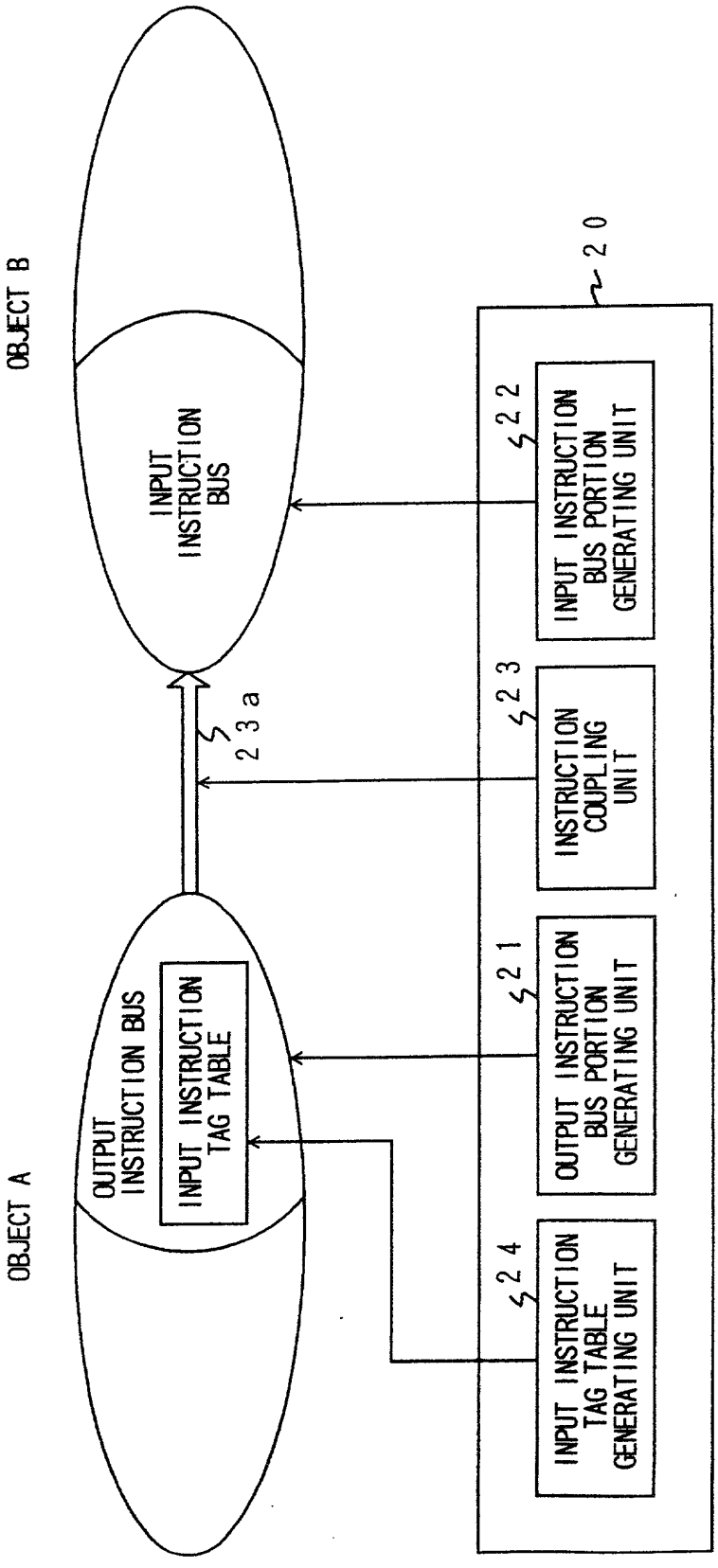

OBJECT A
OBJECT B
OUTPUT INSTRUCTION BUS
INPUT INSTRUCTION TAG TABLE
INPUT INSTRUCTION BUS
23a
20
221
222
223
224
OUTPUT INSTRUCTION BUS PORTION GENERATING UNIT
INPUT INSTRUCTION BUS PORTION GENERATING UNIT
INSTRUCTION COUPLING UNIT
INPUT INSTRUCTION TAG TABLE GENERATING UNIT

Fig. 16

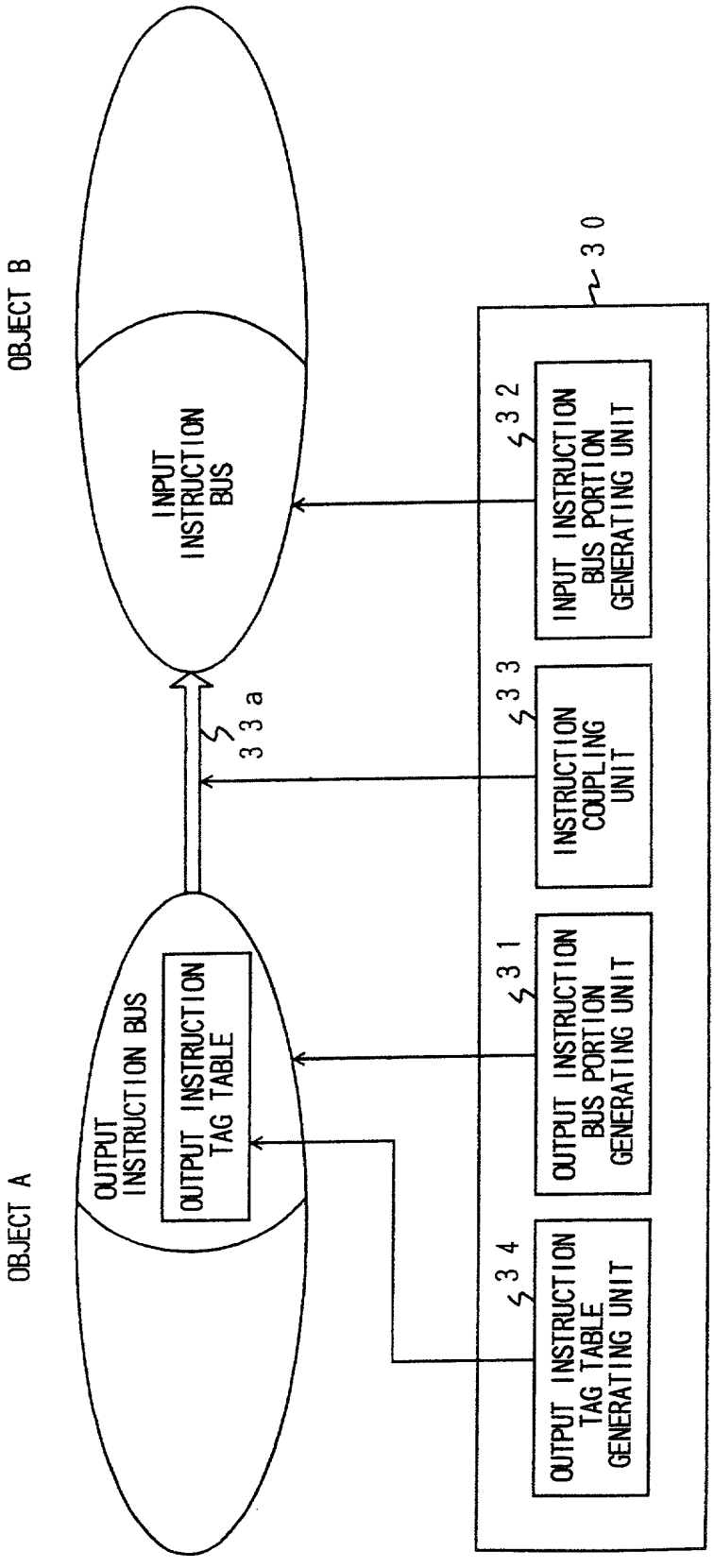

OBJECT A
OUTPUT INSTRUCTION BUS
OUTPUT INSTRUCTION TAG TABLE
OBJECT B
INPUT INSTRUCTION BUS
333a
30
331
OUTPUT INSTRUCTION BUS PORTION GENERATING UNIT
332
INPUT INSTRUCTION BUS PORTION GENERATING UNIT
333
INSTRUCTION COUPLING UNIT
334
OUTPUT INSTRUCTION TAG TABLE GENERATING UNIT

Fig. 17

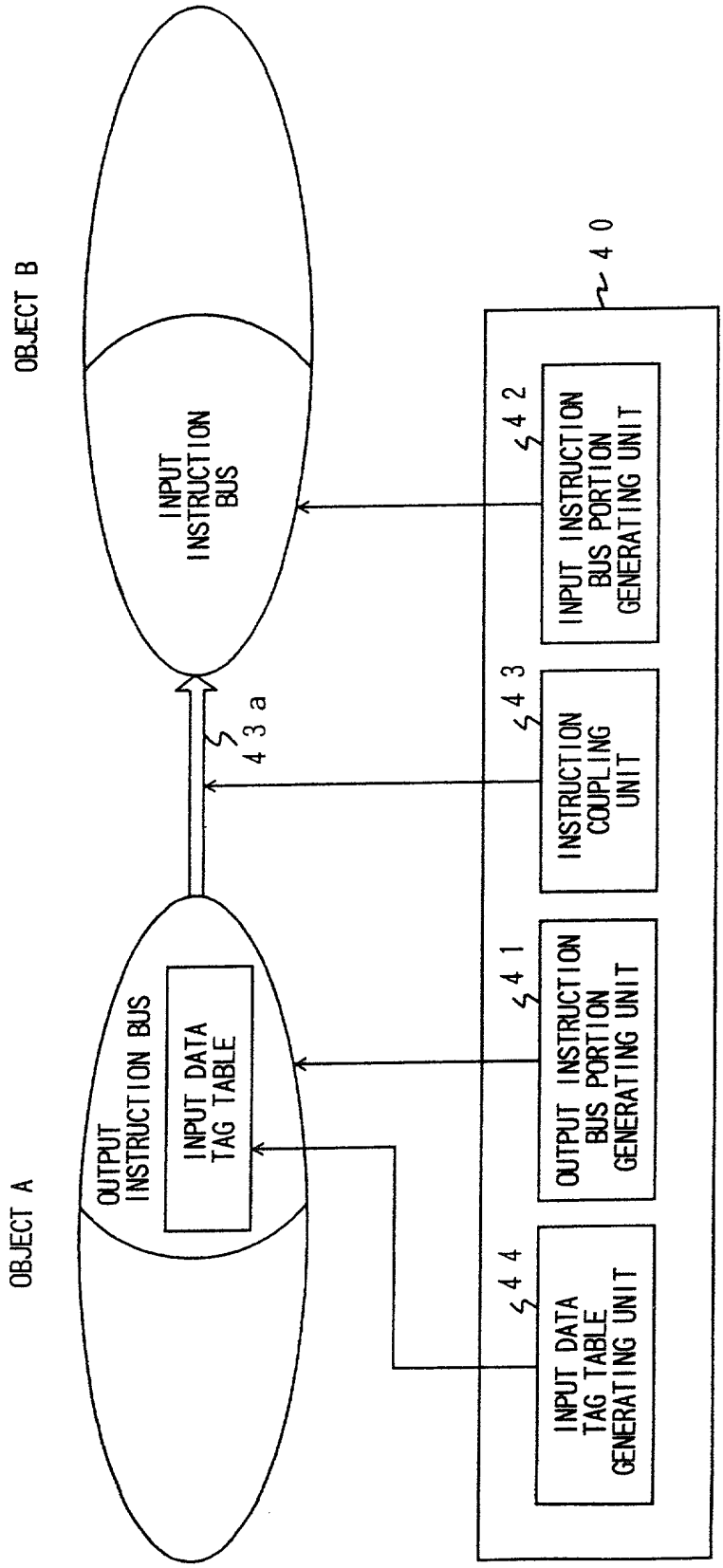

OBJECT A
OUTPUT INSTRUCTION BUS
INPUT DATA TAG TABLE
OBJECT B
INPUT INSTRUCTION BUS
43a
40
41
OUTPUT INSTRUCTION BUS PORTION GENERATING UNIT
42
INPUT INSTRUCTION BUS PORTION GENERATING UNIT
43
INSTRUCTION COUPLING UNIT
44
INPUT DATA TAG TABLE GENERATING UNIT

Fig. 18

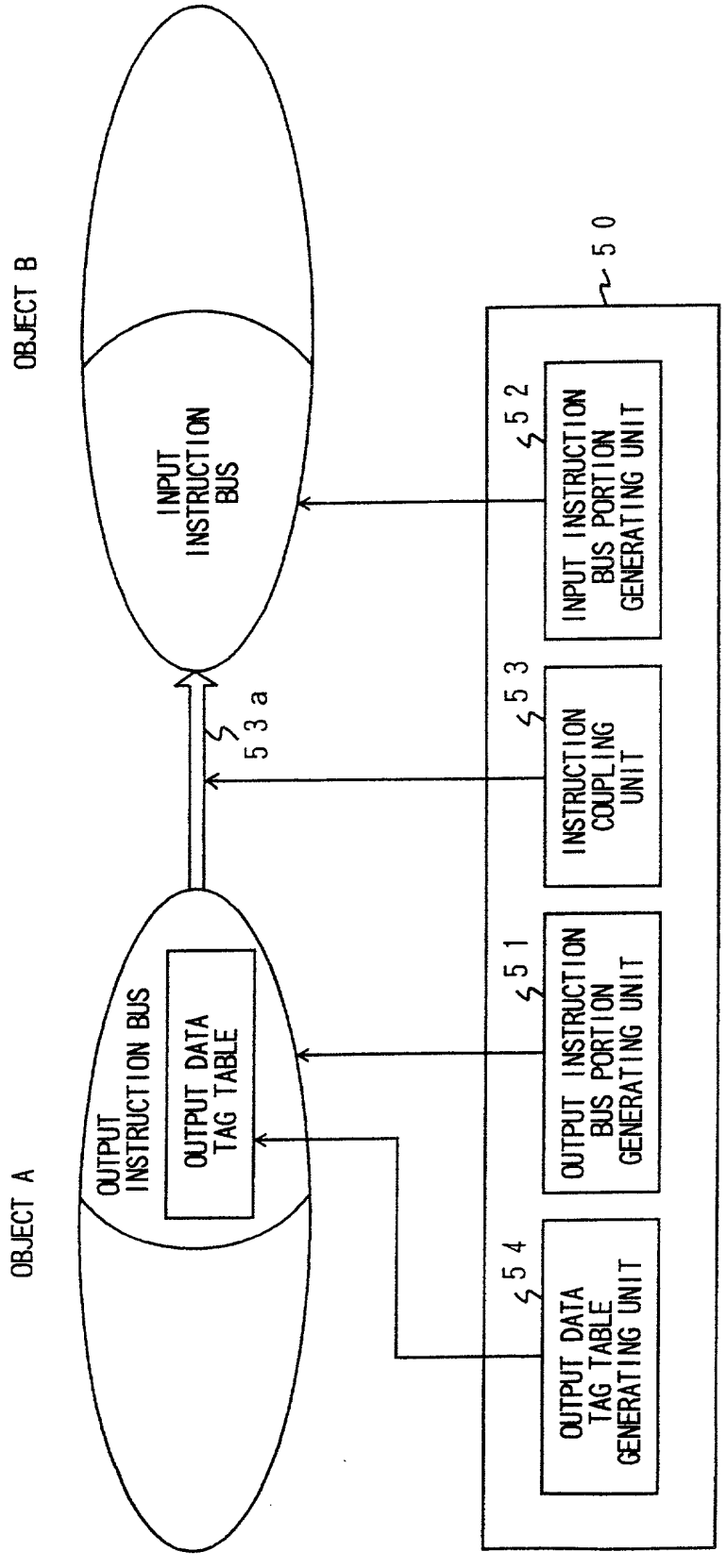

OBJECT A
OBJECT B
OUTPUT INSTRUCTION BUS
OUTPUT DATA TAG TABLE
INPUT INSTRUCTION BUS
53a
50
51
OUTPUT INSTRUCTION BUS PORTION GENERATING UNIT
52
INPUT INSTRUCTION BUS PORTION GENERATING UNIT
53
INSTRUCTION COUPLING UNIT
54
OUTPUT DATA TAG TABLE GENERATING UNIT

Fig. 19

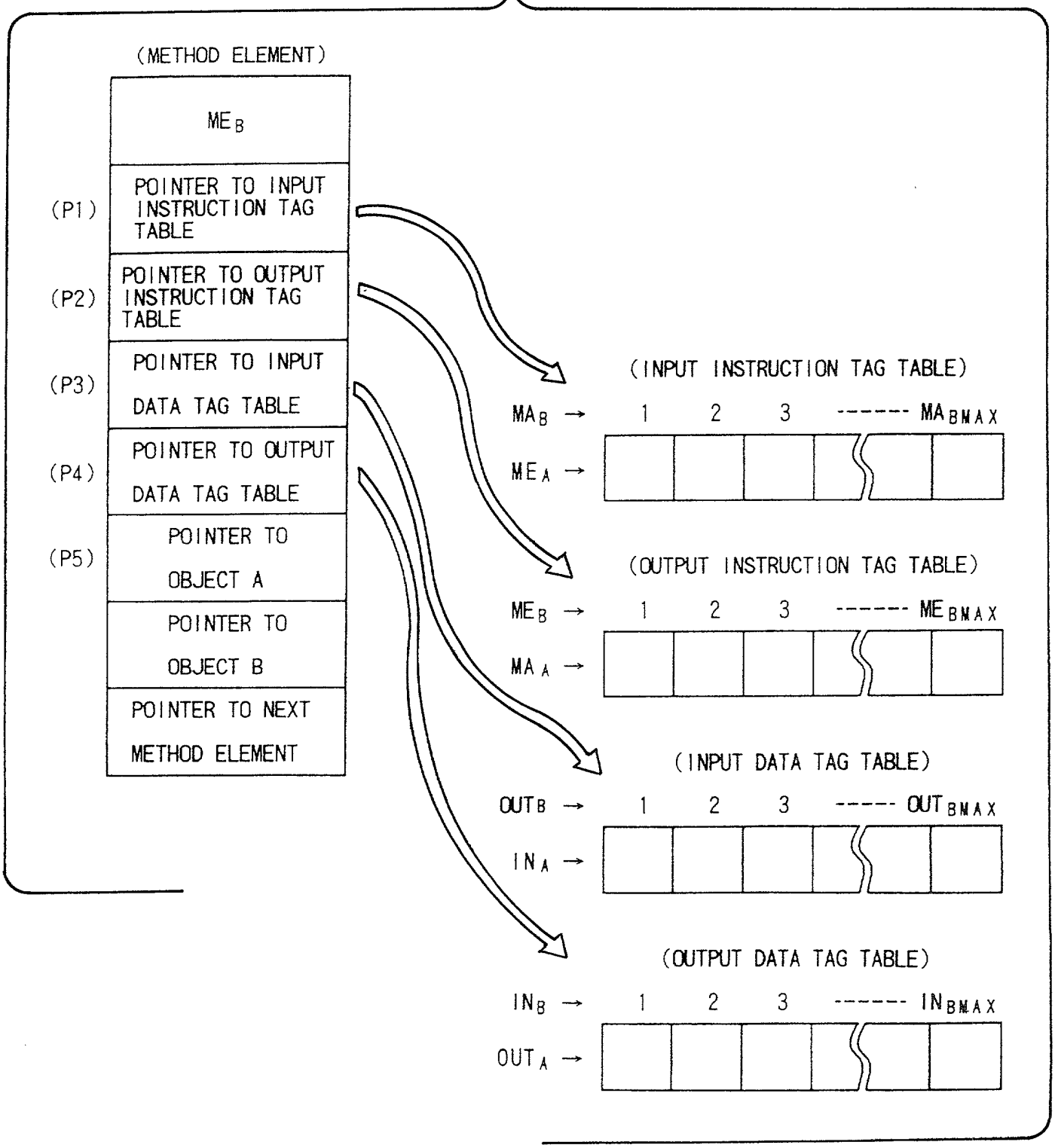

(METHOD ELEMENT)
ME B
(P1)
POINTER TO INPUT INSTRUCTION TAG TABLE
(P2)
POINTER TO OUTPUT INSTRUCTION TAG TABLE
(P3)
POINTER TO INPUT DATA TAG TABLE
(P4)
POINTER TO OUTPUT DATA TAG TABLE
(P5)
POINTER TO OBJECT A
POINTER TO OBJECT B
POINTER TO NEXT METHOD ELEMENT
(INPUT INSTRUCTION TAG TABLE)
MA B →
ME A →
1 2 3 ------ MA BMAX
(OUTPUT INSTRUCTION TAG TABLE)
ME B →
MA A →
1 2 3 ------ ME BMAX
(INPUT DATA TAG TABLE)
OUT B →
IN A →
1 2 3 ----- OUT BMAX
(OUTPUT DATA TAG TABLE)
IN B →
OUT A →
1 2 3 ------ IN BMAX

Fig. 20

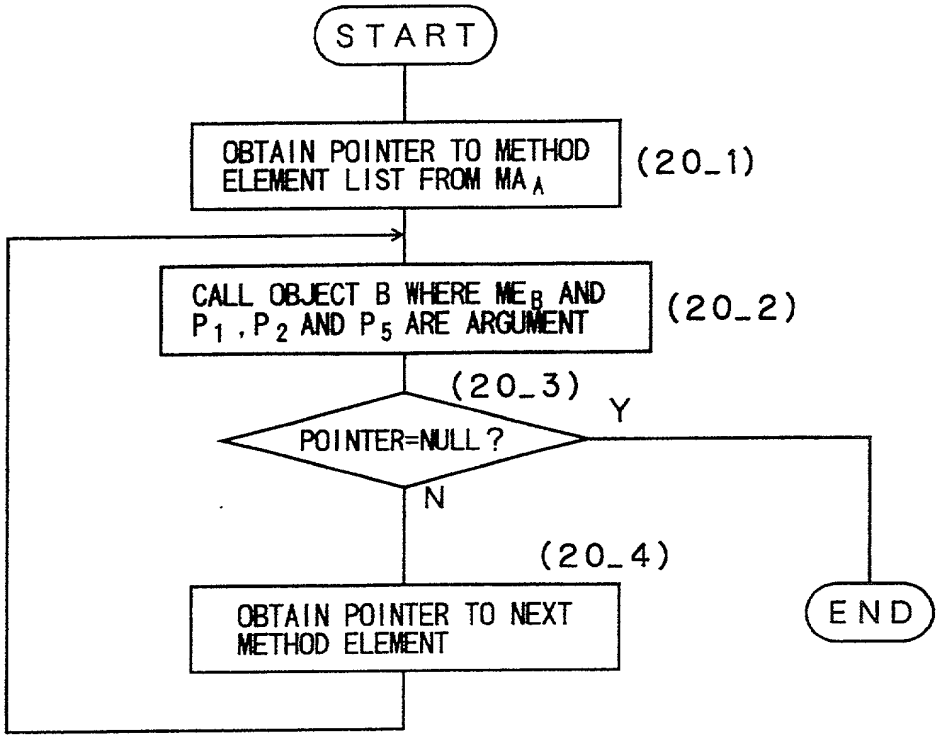

START
OBTAIN POINTER TO METHOD ELEMENT LIST FROM MA_A
(20_1)
CALL OBJECT B WHERE ME_B AND P_1, P_2 AND P_5 ARE ARGUMENT
(20_2)
(20_3)
POINTER=NULL?
Y
N
(20_4)
OBTAIN POINTER TO NEXT METHOD ELEMENT
END

Fig. 21

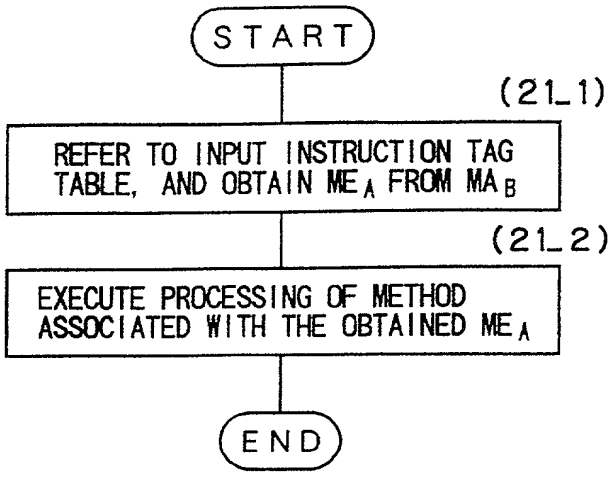

START
(21_1)
REFER TO INPUT INSTRUCTION TAG TABLE, AND OBTAIN ME_A FROM MA_B
(21_2)
EXECUTE PROCESSING OF METHOD ASSOCIATED WITH THE OBTAINED ME_A
END

Fig.22

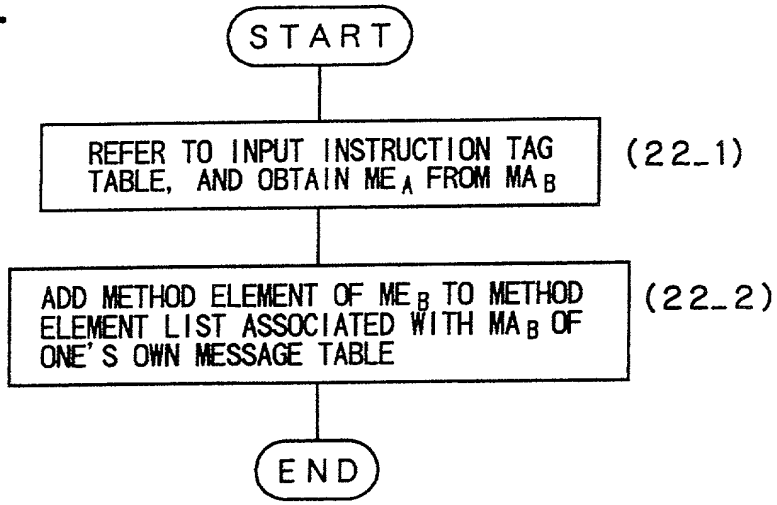


Fig.23

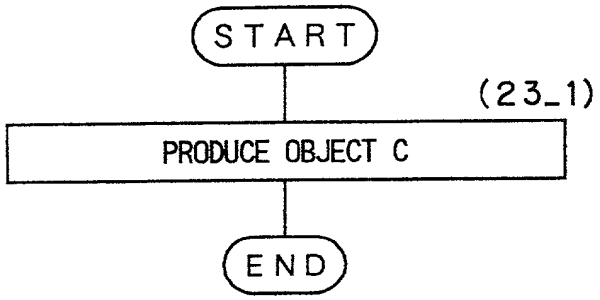


Fig.24

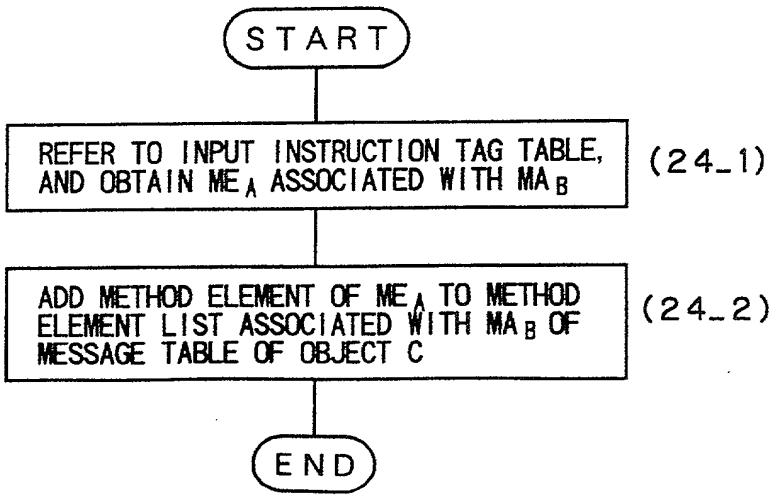


Fig. 25

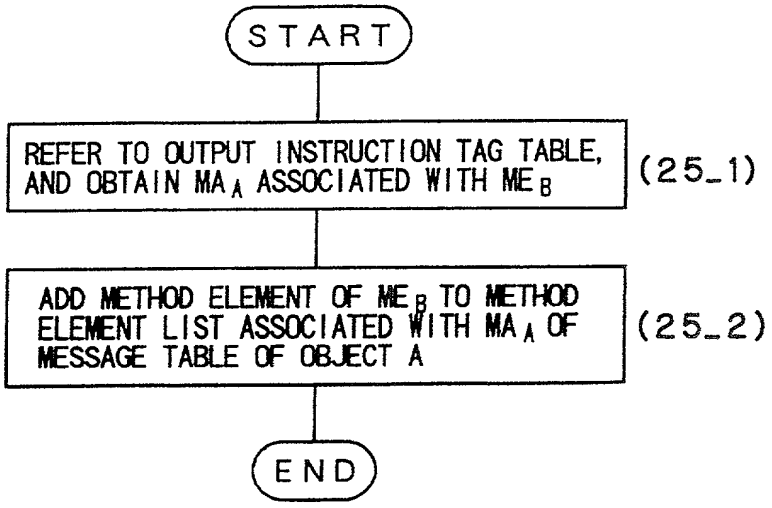

START
REFER TO OUTPUT INSTRUCTION TAG TABLE, AND OBTAIN MA_A ASSOCIATED WITH ME_B
(25_1)
ADD METHOD ELEMENT OF ME_B TO METHOD ELEMENT LIST ASSOCIATED WITH MA_A OF MESSAGE TABLE OF OBJECT A
(25_2)
END

Fig. 26

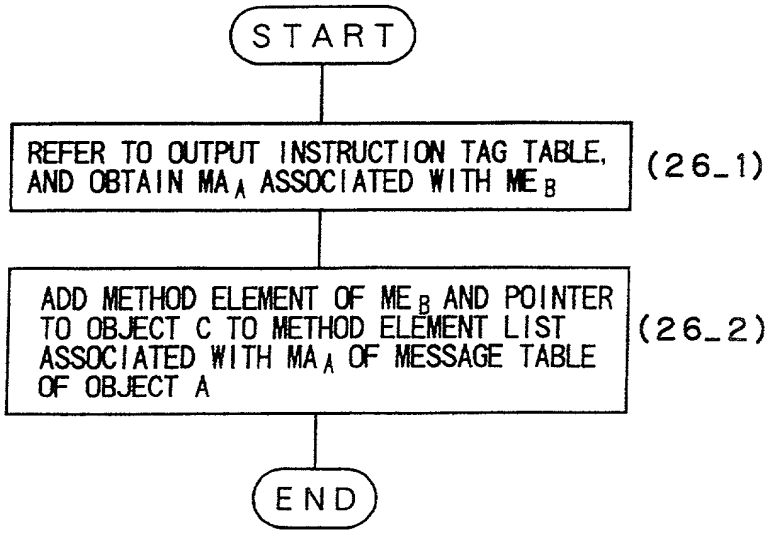

START
REFER TO OUTPUT INSTRUCTION TAG TABLE, AND OBTAIN MA_A ASSOCIATED WITH ME_B
(26_1)
ADD METHOD ELEMENT OF ME_B AND POINTER TO OBJECT C TO METHOD ELEMENT LIST ASSOCIATED WITH MA_A OF MESSAGE TABLE OF OBJECT A
(26_2)
END

Fig. 27

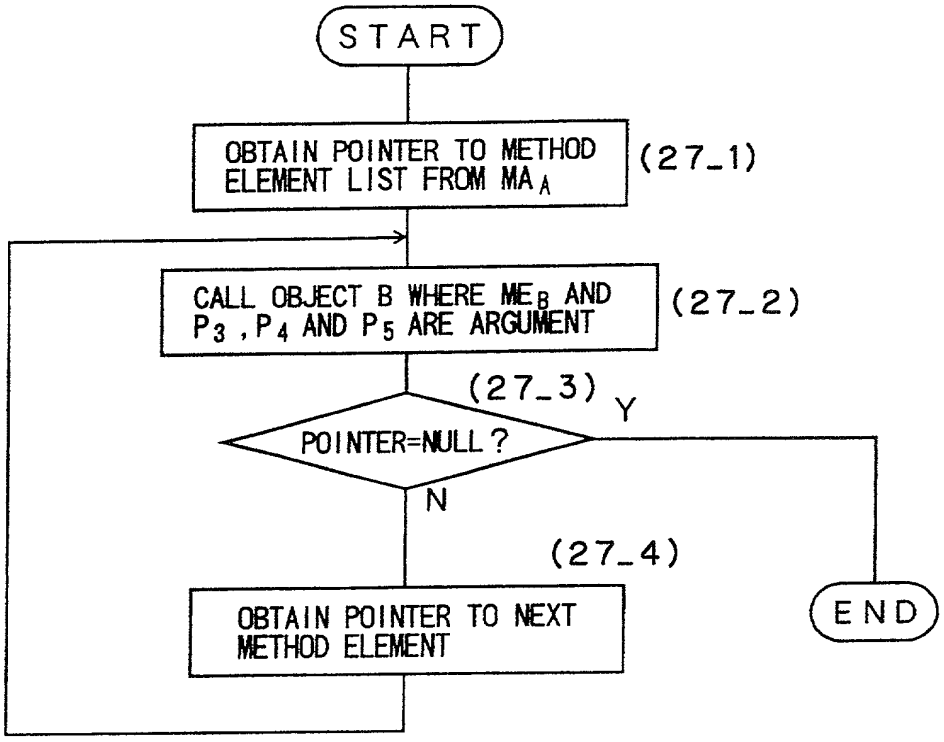

START
OBTAIN POINTER TO METHOD ELEMENT LIST FROM MA_A
(27_1)
CALL OBJECT B WHERE ME_B AND P_3, P_4 AND P_5 ARE ARGUMENT
(27_2)
(27_3)
POINTER=NULL?
Y
N
(27_4)
OBTAIN POINTER TO NEXT METHOD ELEMENT
END

Fig. 28

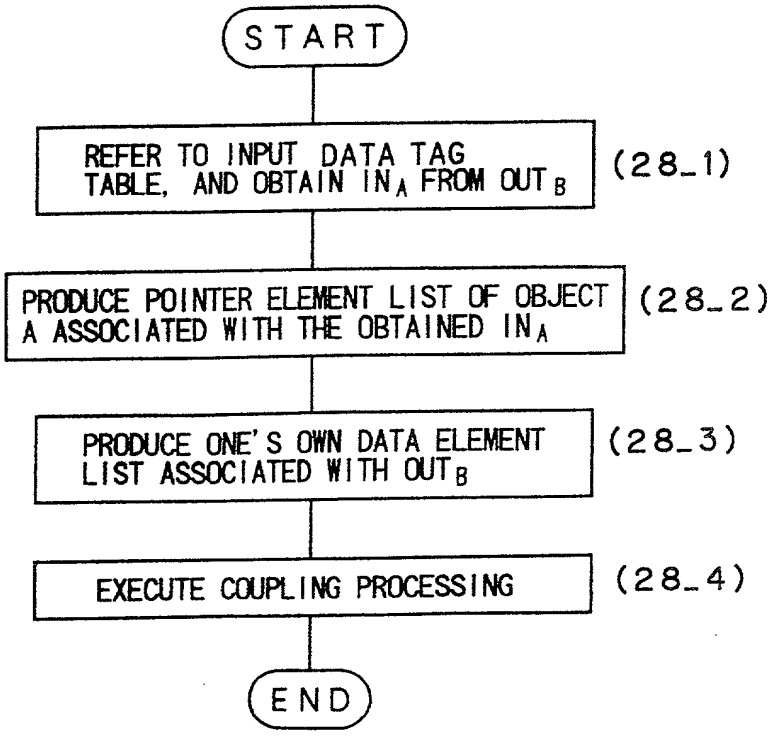

START
REFER TO INPUT DATA TAG TABLE, AND OBTAIN IN_A FROM OUT_B
(28_1)
PRODUCE POINTER ELEMENT LIST OF OBJECT A ASSOCIATED WITH THE OBTAINED IN_A
(28_2)
PRODUCE ONE'S OWN DATA ELEMENT LIST ASSOCIATED WITH OUT_B
(28_3)
EXECUTE COUPLING PROCESSING
(28_4)
END

Fig. 29

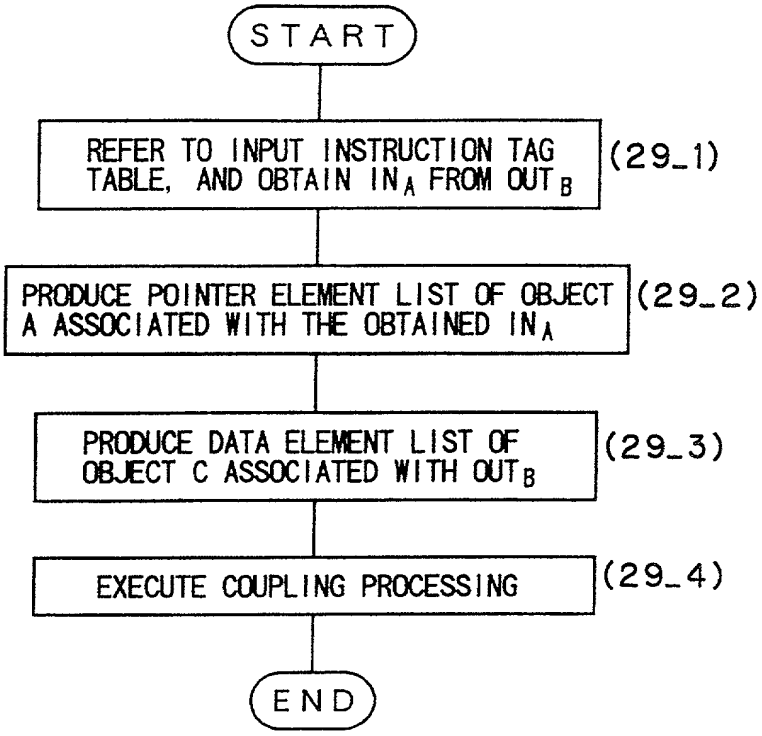

START
REFER TO INPUT INSTRUCTION TAG TABLE, AND OBTAIN IN_A FROM OUT_B
(29_1)
PRODUCE POINTER ELEMENT LIST OF OBJECT A ASSOCIATED WITH THE OBTAINED IN_A
(29_2)
PRODUCE DATA ELEMENT LIST OF OBJECT C ASSOCIATED WITH OUT_B
(29_3)
EXECUTE COUPLING PROCESSING
(29_4)
END

Fig. 30

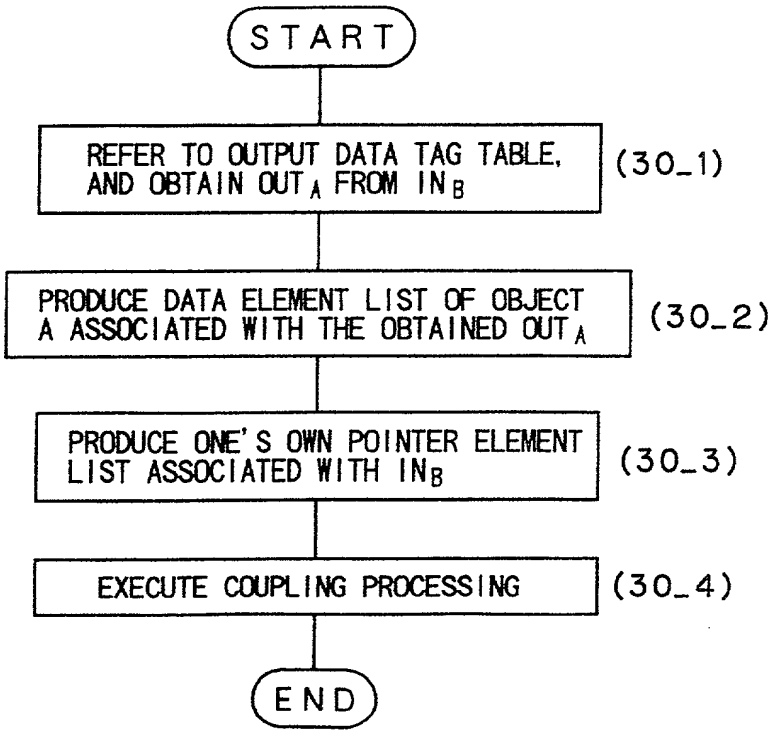

START
REFER TO OUTPUT DATA TAG TABLE, AND OBTAIN OUT_A FROM IN_B
(30_1)
PRODUCE DATA ELEMENT LIST OF OBJECT A ASSOCIATED WITH THE OBTAINED OUT_A
(30_2)
PRODUCE ONE'S OWN POINTER ELEMENT LIST ASSOCIATED WITH IN_B
(30_3)
EXECUTE COUPLING PROCESSING
(30_4)
END

Fig. 31

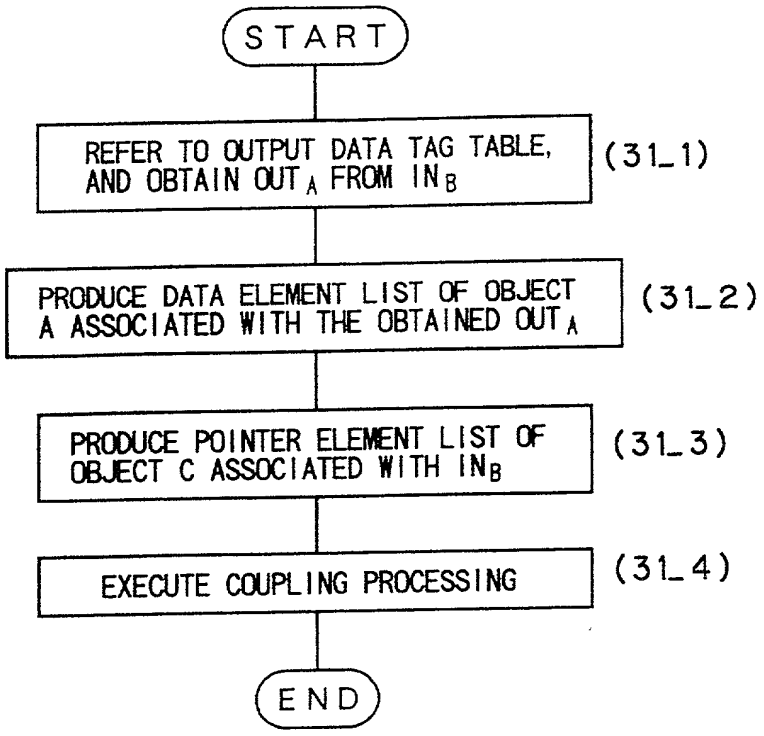

START
REFER TO OUTPUT DATA TAG TABLE, AND OBTAIN OUT$_A$ FROM IN$_B$
(31_1)
PRODUCE DATA ELEMENT LIST OF OBJECT A ASSOCIATED WITH THE OBTAINED OUT$_A$
(31_2)
PRODUCE POINTER ELEMENT LIST OF OBJECT C ASSOCIATED WITH IN$_B$
(31_3)
EXECUTE COUPLING PROCESSING
(31_4)
END

Fig. 32

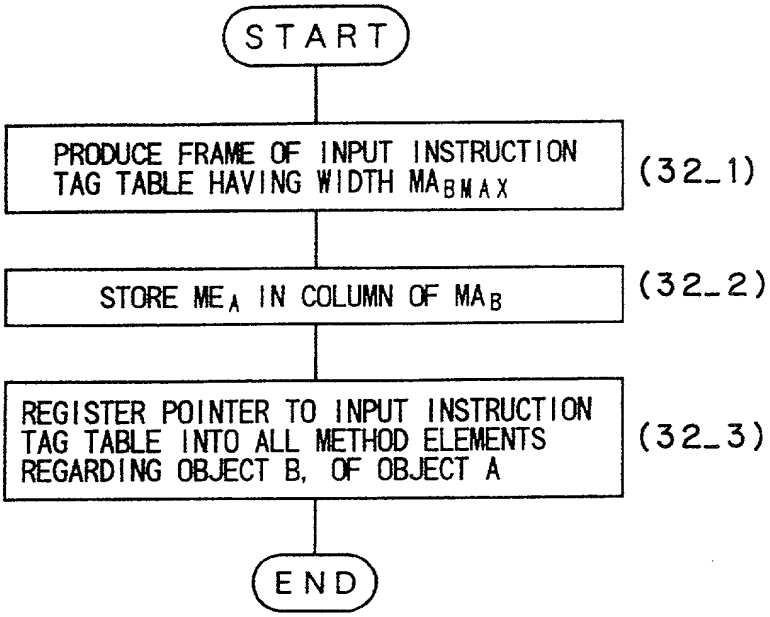

START
PRODUCE FRAME OF INPUT INSTRUCTION TAG TABLE HAVING WIDTH MA$_{BMAX}$
(32_1)
STORE ME$_A$ IN COLUMN OF MA$_B$
(32_2)
REGISTER POINTER TO INPUT INSTRUCTION TAG TABLE INTO ALL METHOD ELEMENTS REGARDING OBJECT B, OF OBJECT A
(32_3)
END

Fig. 33

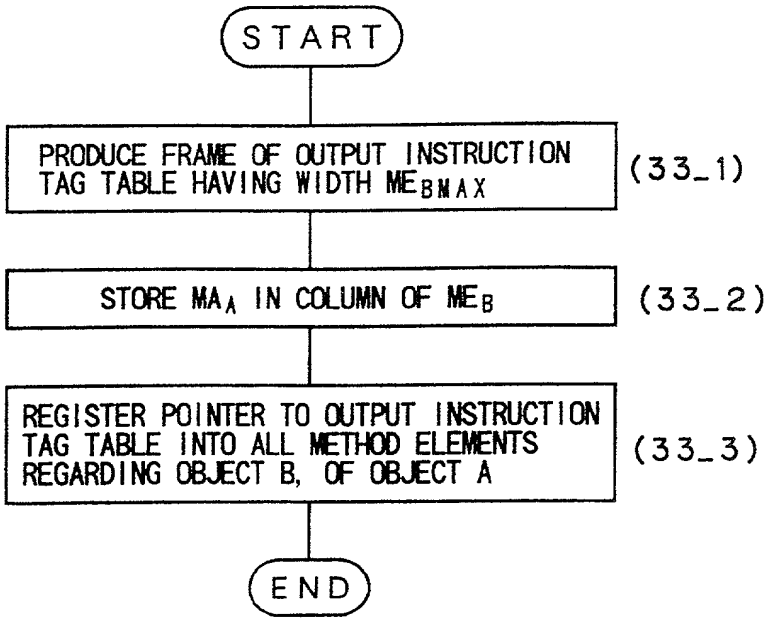

START
PRODUCE FRAME OF OUTPUT INSTRUCTION TAG TABLE HAVING WIDTH ME_{BMAX}
(33_1)
STORE MA_A IN COLUMN OF ME_B
(33_2)
REGISTER POINTER TO OUTPUT INSTRUCTION TAG TABLE INTO ALL METHOD ELEMENTS REGARDING OBJECT B, OF OBJECT A
(33_3)
END

Fig. 34

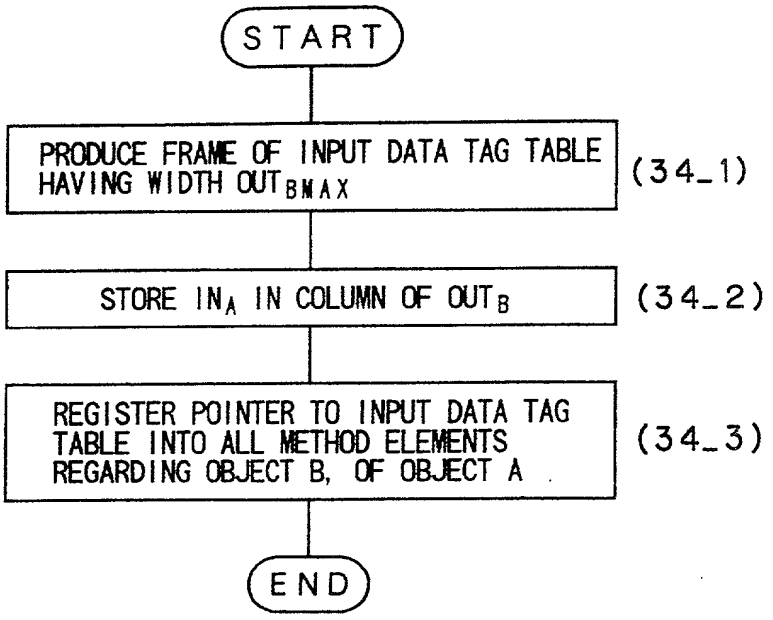

START
PRODUCE FRAME OF INPUT DATA TAG TABLE HAVING WIDTH OUT_{BMAX}
(34_1)
STORE IN_A IN COLUMN OF OUT_B
(34_2)
REGISTER POINTER TO INPUT DATA TAG TABLE INTO ALL METHOD ELEMENTS REGARDING OBJECT B, OF OBJECT A
(34_3)
END

Fig. 35

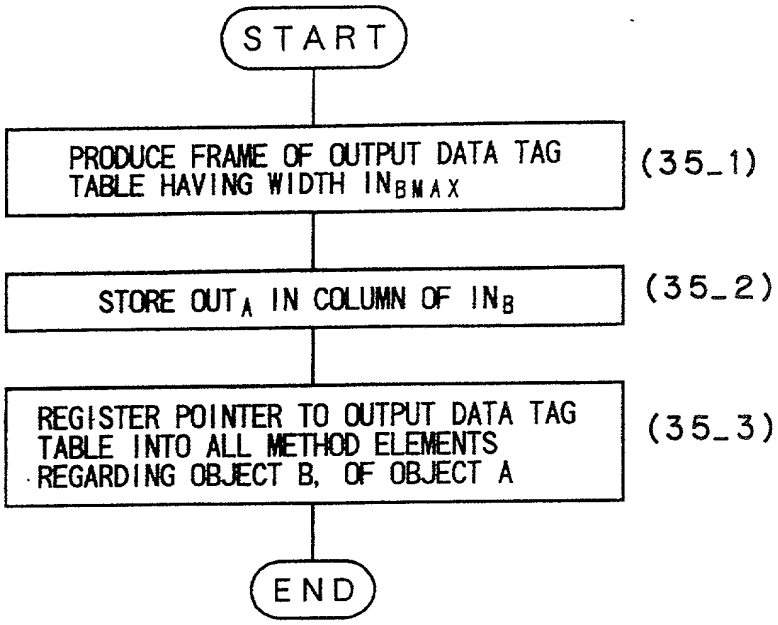

START
PRODUCE FRAME OF OUTPUT DATA TAG TABLE HAVING WIDTH IN_{BMAX}
(35_1)
STORE OUT_A IN COLUMN OF IN_B
(35_2)
REGISTER POINTER TO OUTPUT DATA TAG TABLE INTO ALL METHOD ELEMENTS REGARDING OBJECT B, OF OBJECT A
(35_3)
END

Fig. 36

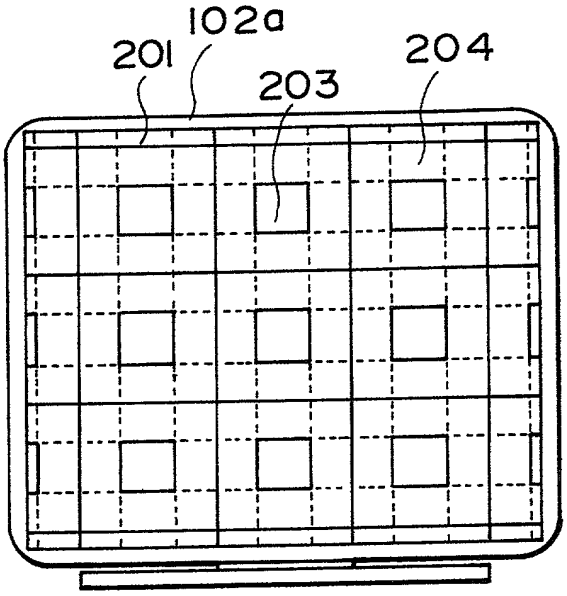


Fig. 37

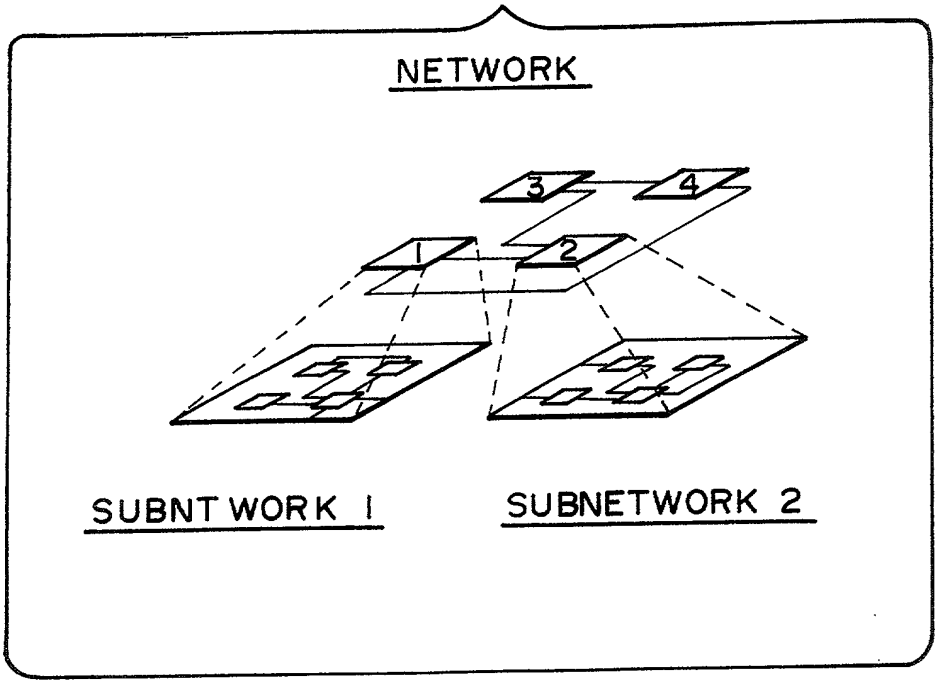


Fig.38(A)

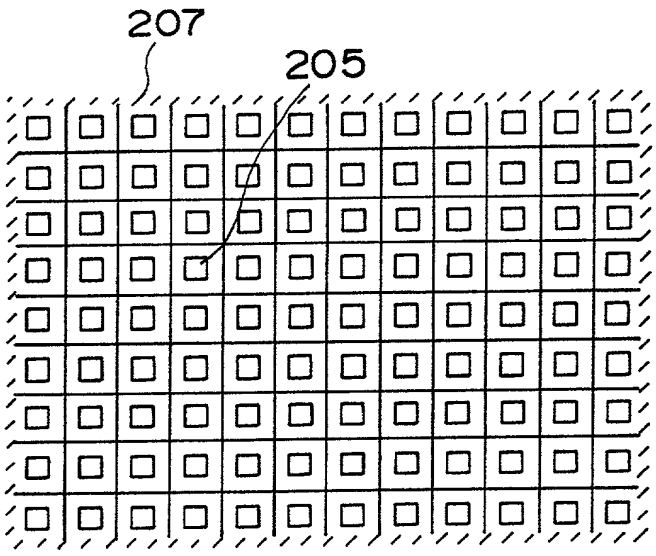


Fig.38(B)

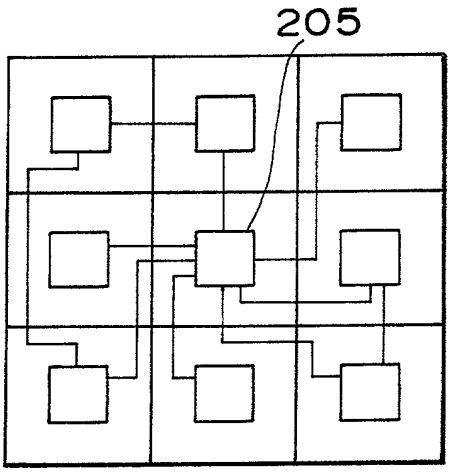


Fig.39(A)

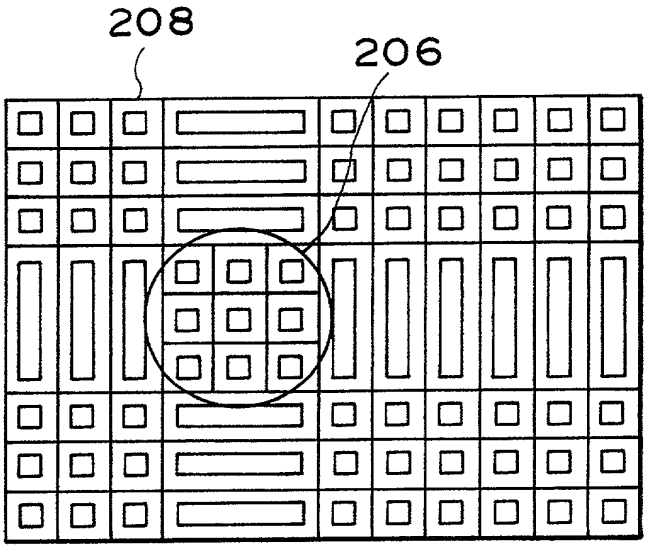
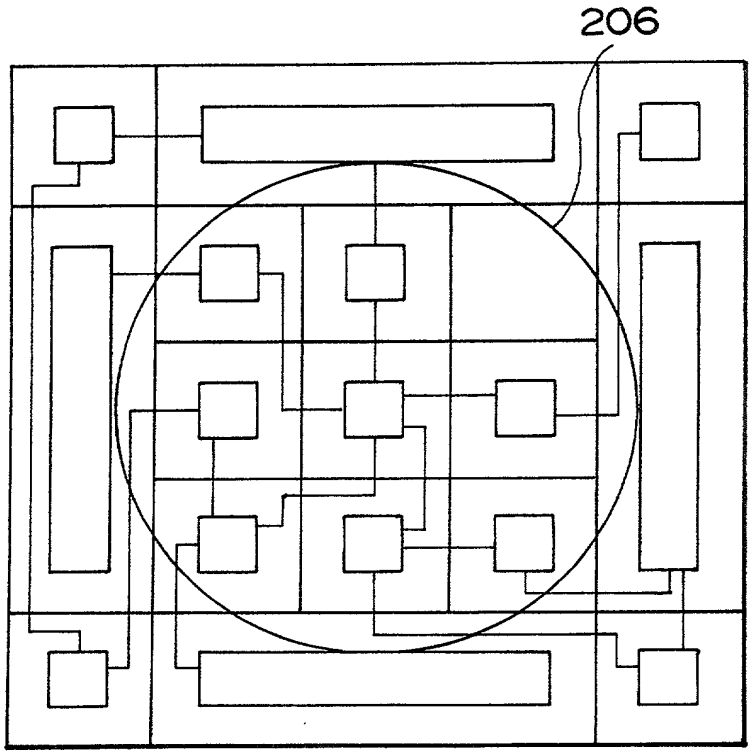


Fig.39(B)



JP2003045950 = 2003-02-14 05/09/50

Fig.40(A)

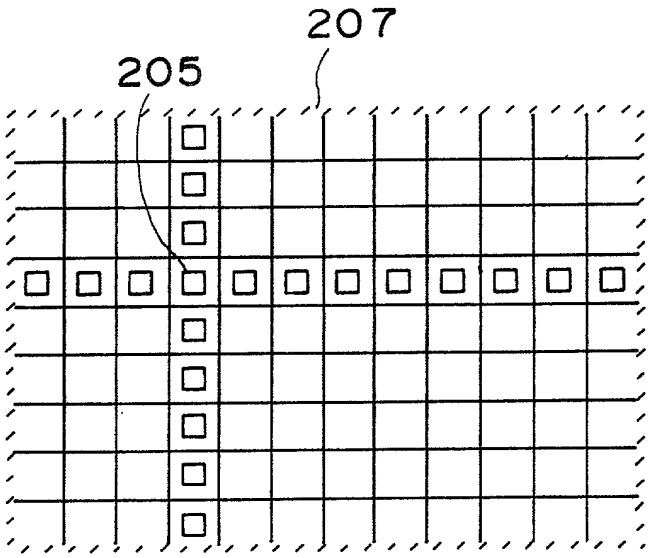


Fig.40(B)

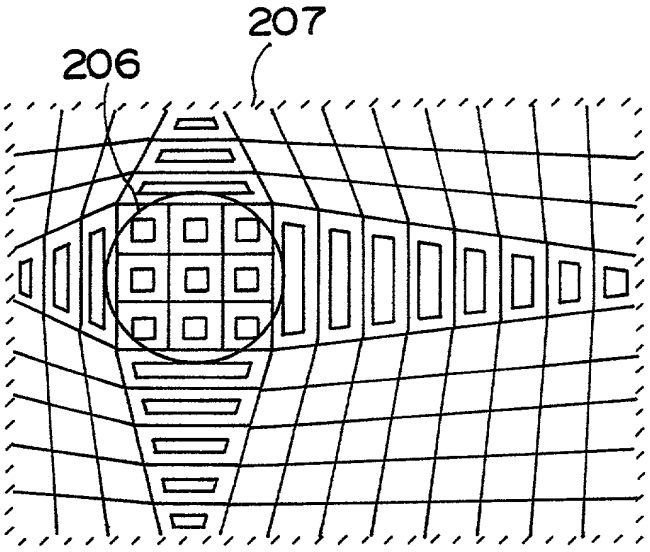


Fig. 41(B)

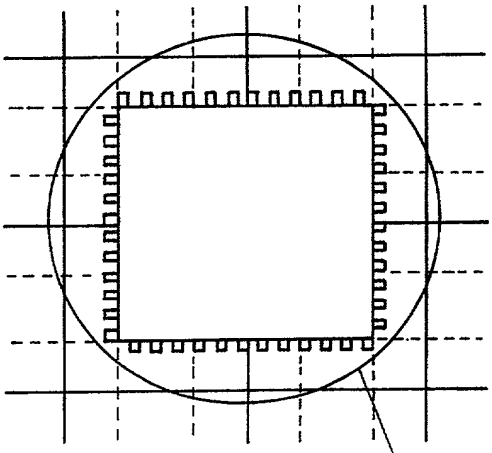


Fig. 41(C)

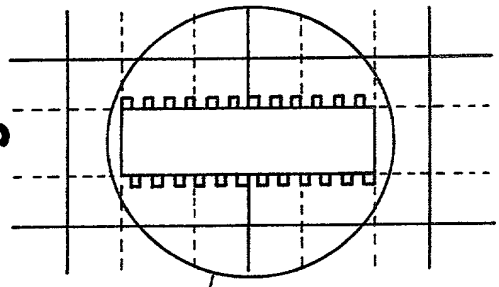


Fig. 41(A)

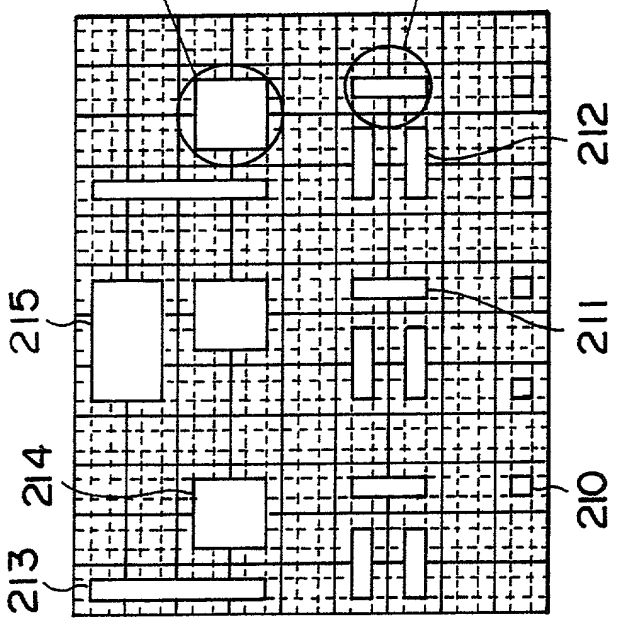


Fig. 42

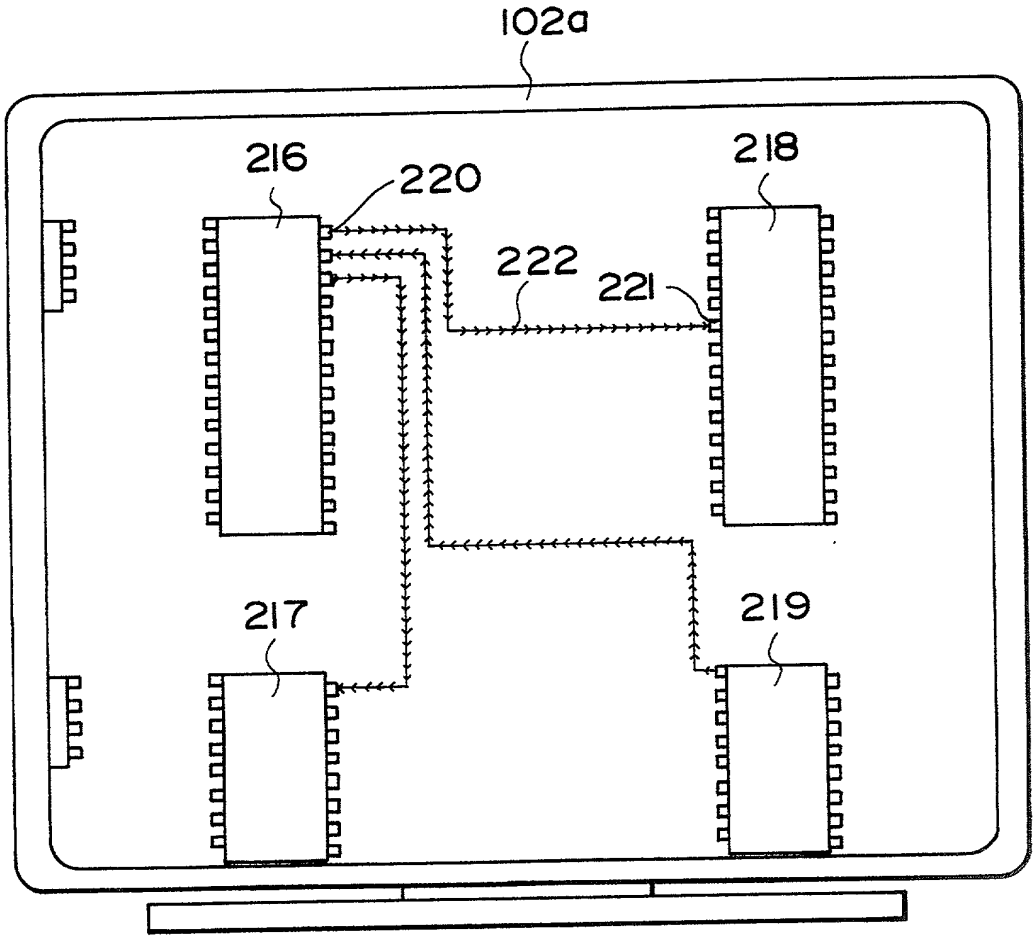


Fig. 43(A)

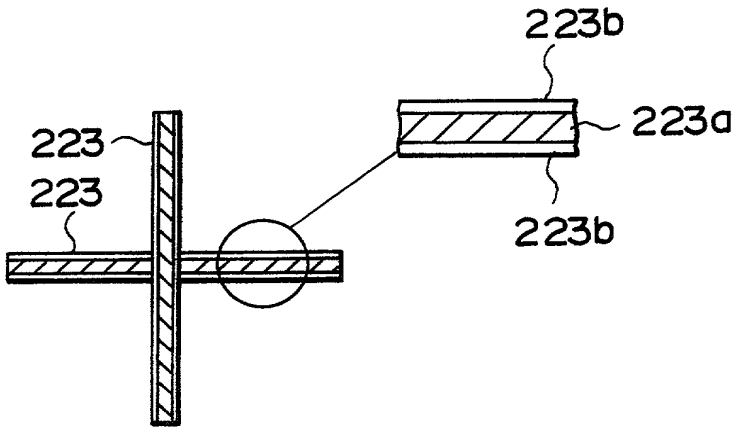


Fig. 43(B)

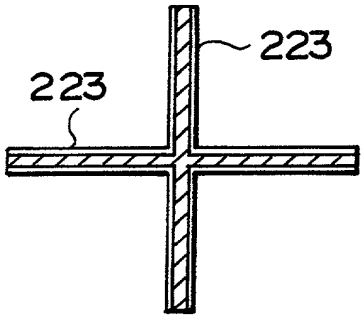


Fig. 44(A)

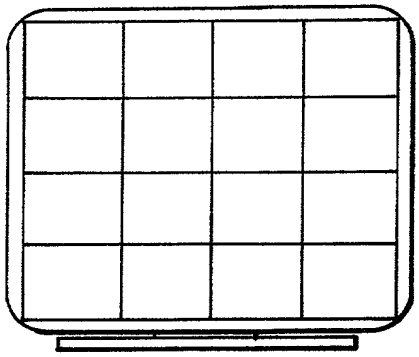


Fig. 44(B)

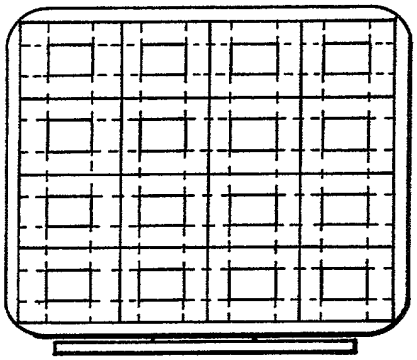


Fig. 44(C)

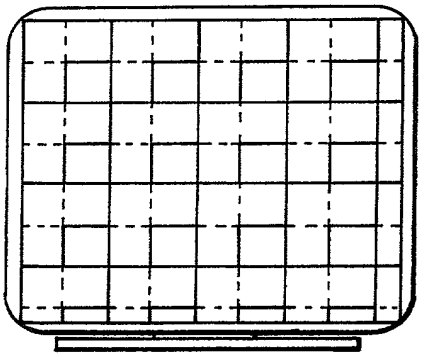


Fig. 45

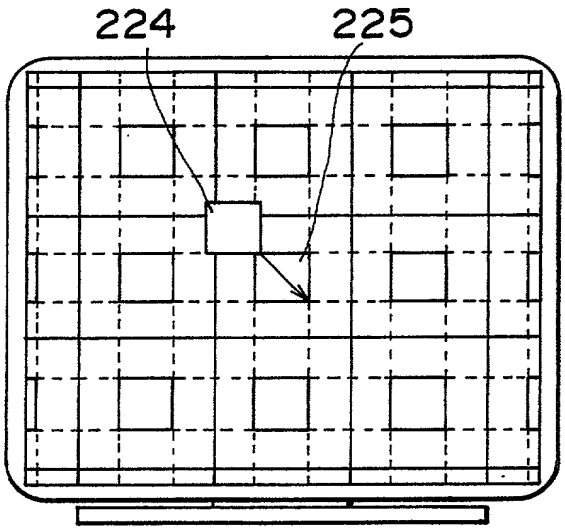


Fig.46(A)

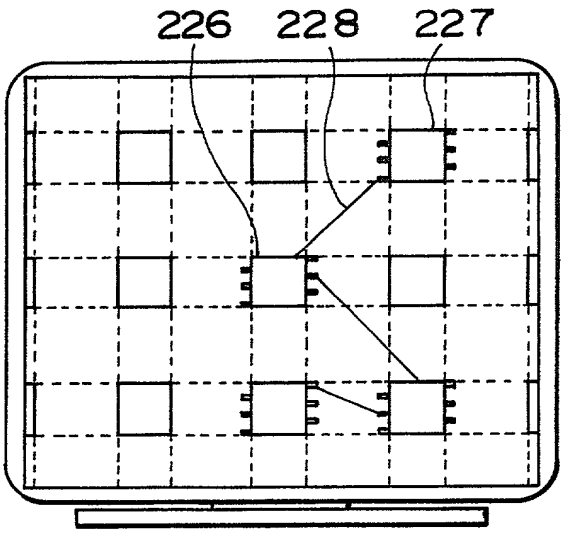


Fig.46(B)

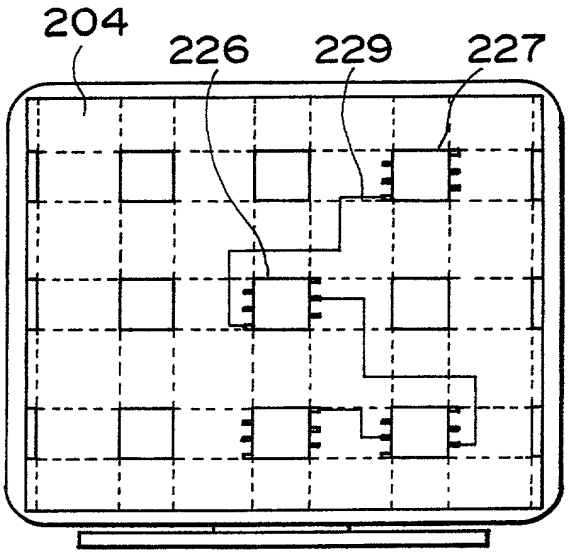


Fig. 47 (A)

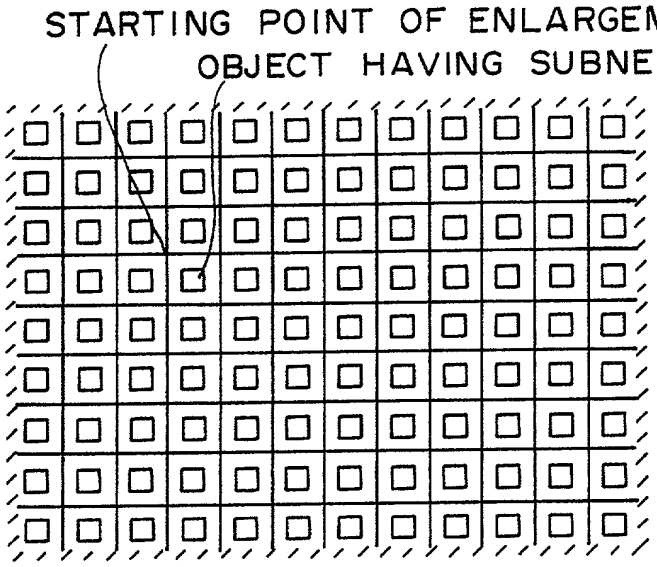


Fig. 47 (B)

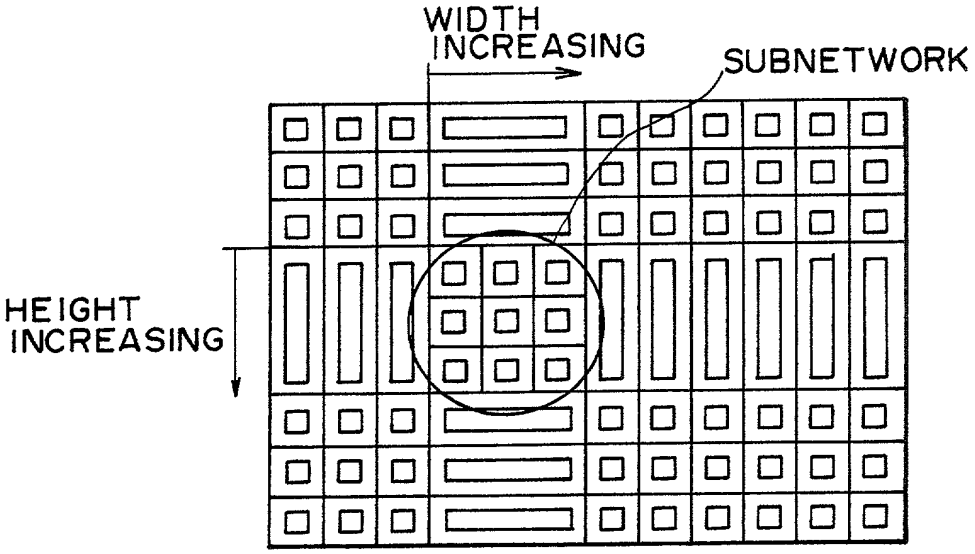


Fig. 48

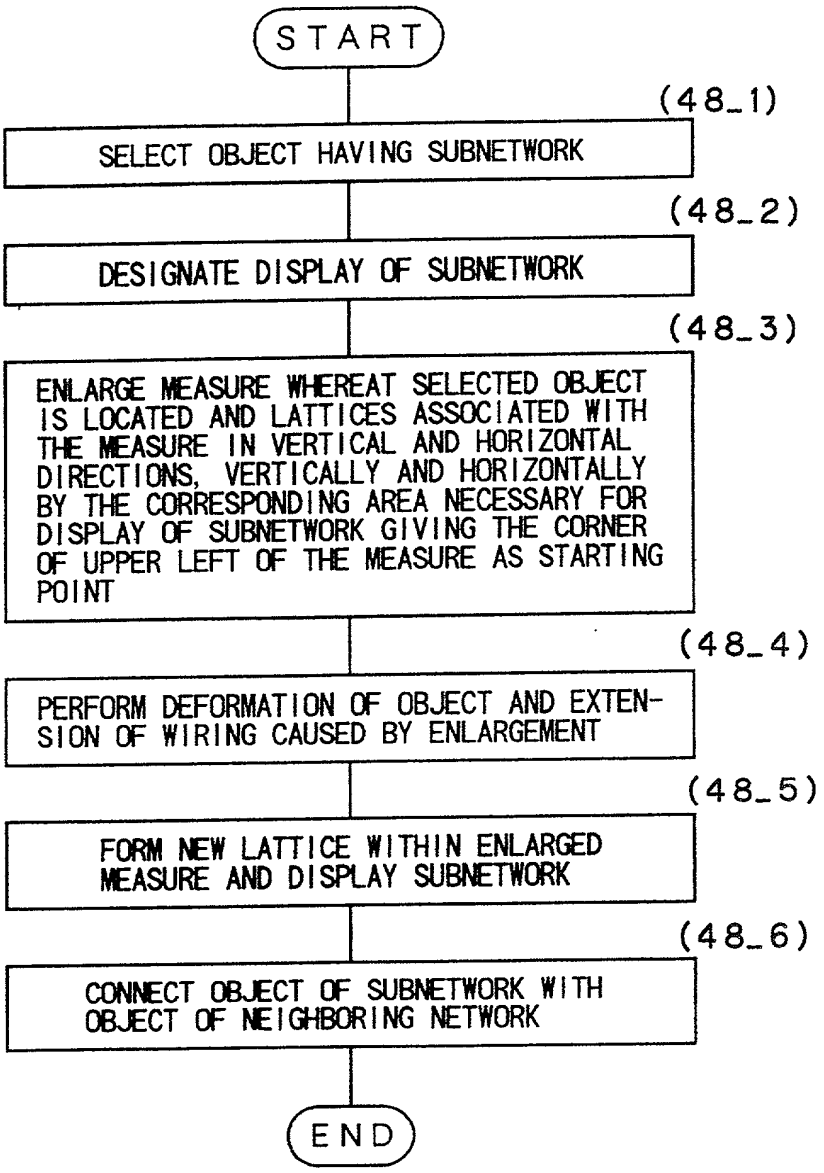

START
(48_1)
SELECT OBJECT HAVING SUBNETWORK
(48_2)
DESIGNATE DISPLAY OF SUBNETWORK
(48_3)
ENLARGE MEASURE WHEREAT SELECTED OBJECT IS LOCATED AND LATTICES ASSOCIATED WITH THE MEASURE IN VERTICAL AND HORIZONTAL DIRECTIONS, VERTICALLY AND HORIZONTALLY BY THE CORRESPONDING AREA NECESSARY FOR DISPLAY OF SUBNETWORK GIVING THE CORNER OF UPPER LEFT OF THE MEASURE AS STARTING POINT
(48_4)
PERFORM DEFORMATION OF OBJECT AND EXTENSION OF WIRING CAUSED BY ENLARGEMENT
(48_5)
FORM NEW LATTICE WITHIN ENLARGED MEASURE AND DISPLAY SUBNETWORK
(48_6)
CONNECT OBJECT OF SUBNETWORK WITH OBJECT OF NEIGHBORING NETWORK
END

Fig.49(A)

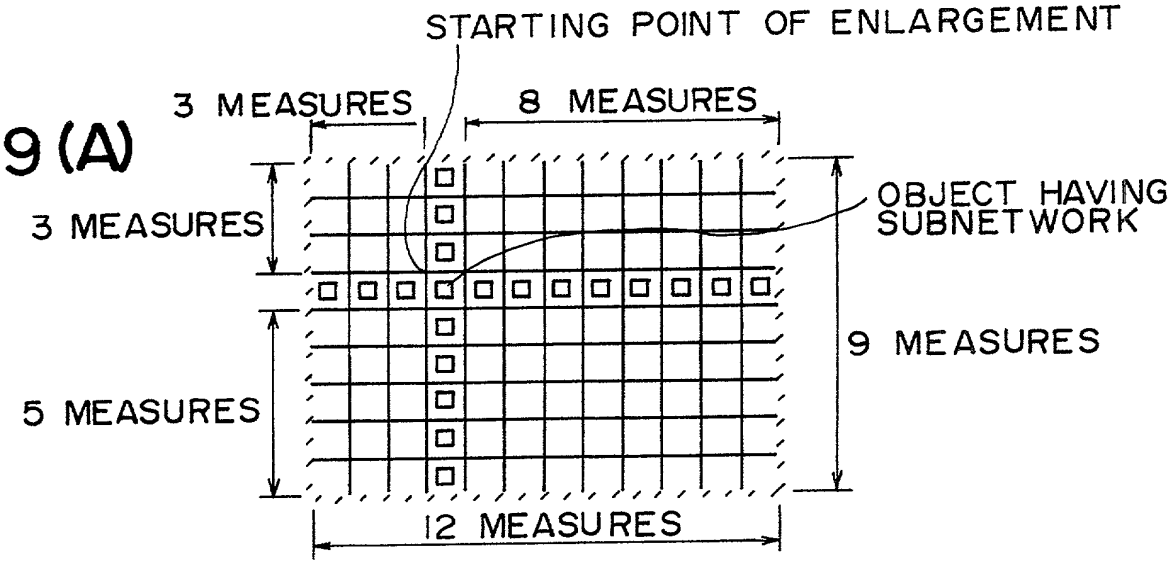


Fig.49(B)

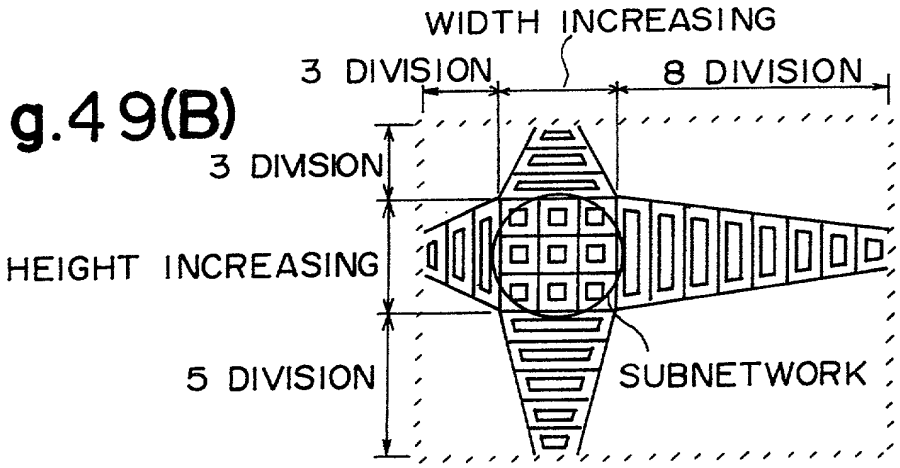


Fig.49(C)

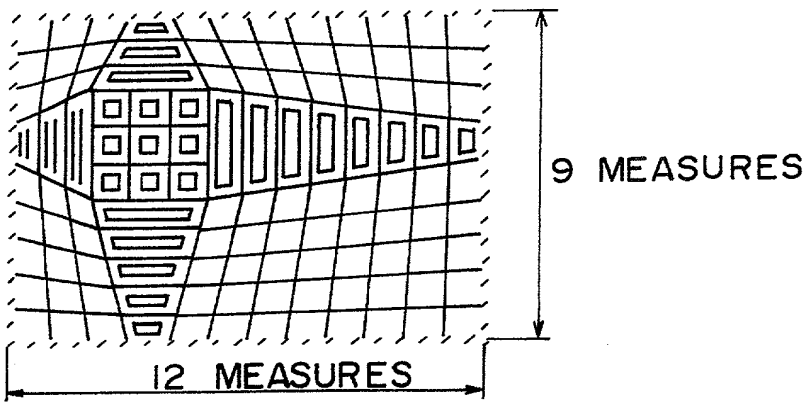


Fig.50

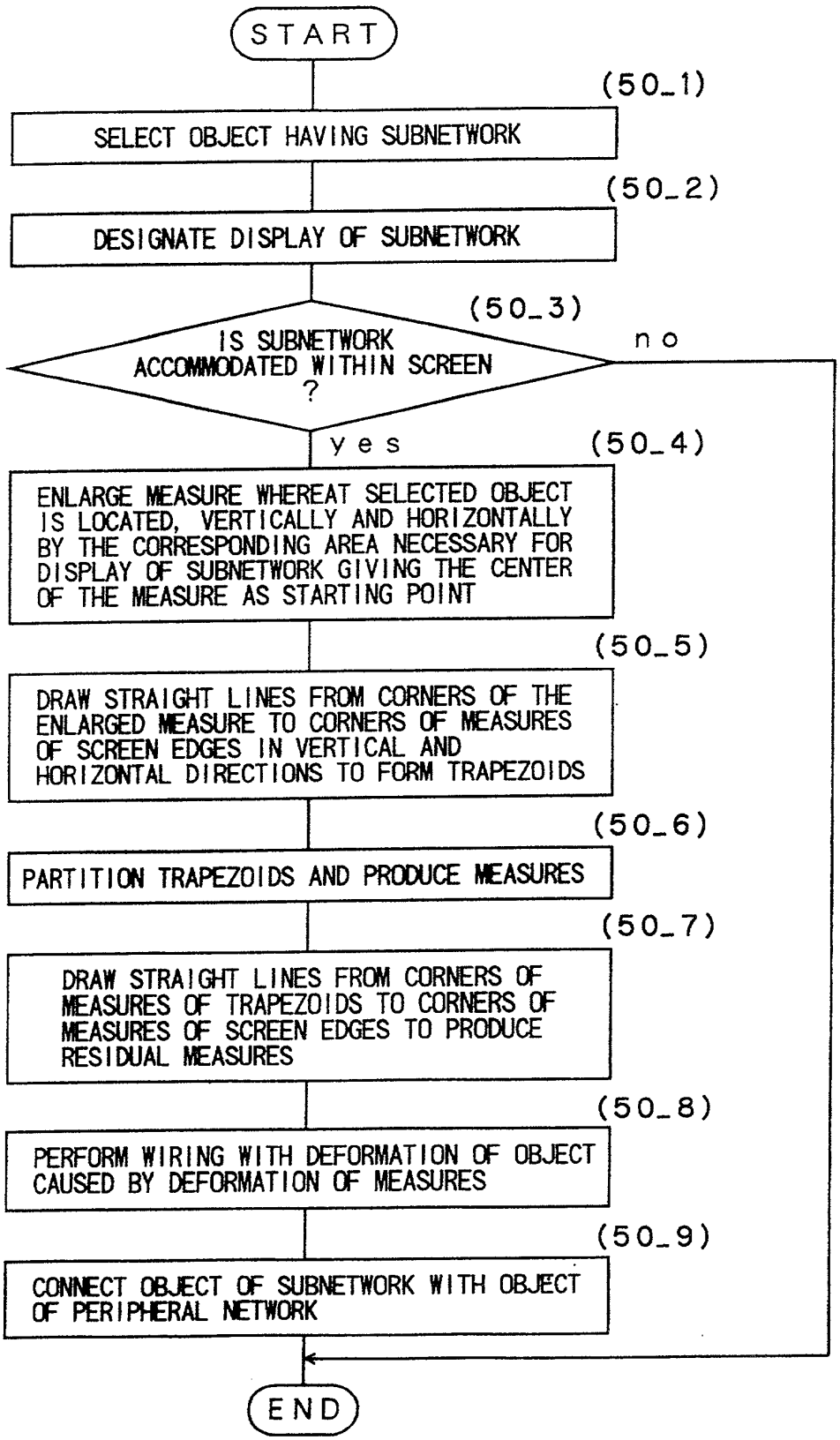

START
(50_1)
SELECT OBJECT HAVING SUBNETWORK
(50_2)
DESIGNATE DISPLAY OF SUBNETWORK
(50_3)
IS SUBNETWORK ACCOMMODATED WITHIN SCREEN ?
no
yes
(50_4)
ENLARGE MEASURE WHEREAT SELECTED OBJECT IS LOCATED, VERTICALLY AND HORIZONTALLY BY THE CORRESPONDING AREA NECESSARY FOR DISPLAY OF SUBNETWORK GIVING THE CENTER OF THE MEASURE AS STARTING POINT
(50_5)
DRAW STRAIGHT LINES FROM CORNERS OF THE ENLARGED MEASURE TO CORNERS OF MEASURES OF SCREEN EDGES IN VERTICAL AND HORIZONTAL DIRECTIONS TO FORM TRAPEZOIDS
(50_6)
PARTITION TRAPEZOIDS AND PRODUCE MEASURES
(50_7)
DRAW STRAIGHT LINES FROM CORNERS OF MEASURES OF TRAPEZOIDS TO CORNERS OF MEASURES OF SCREEN EDGES TO PRODUCE RESIDUAL MEASURES
(50_8)
PERFORM WIRING WITH DEFORMATION OF OBJECT CAUSED BY DEFORMATION OF MEASURES
(50_9)
CONNECT OBJECT OF SUBNETWORK WITH OBJECT OF PERIPHERAL NETWORK
END

Fig.51(A)

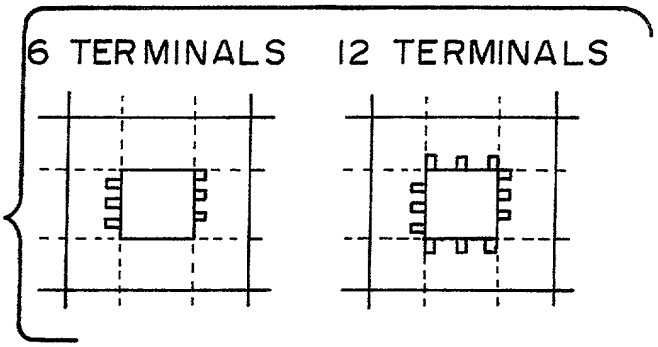


Fig.51(B)

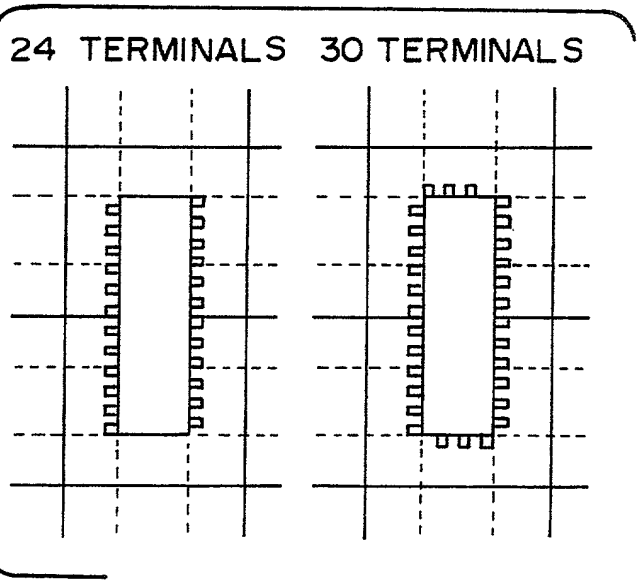


Fig.51(C)

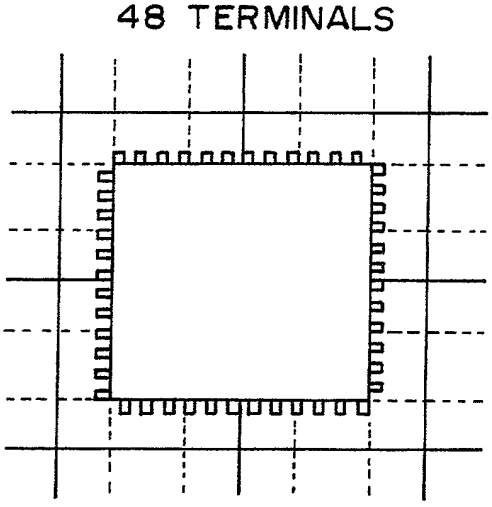


Fig.52(A)

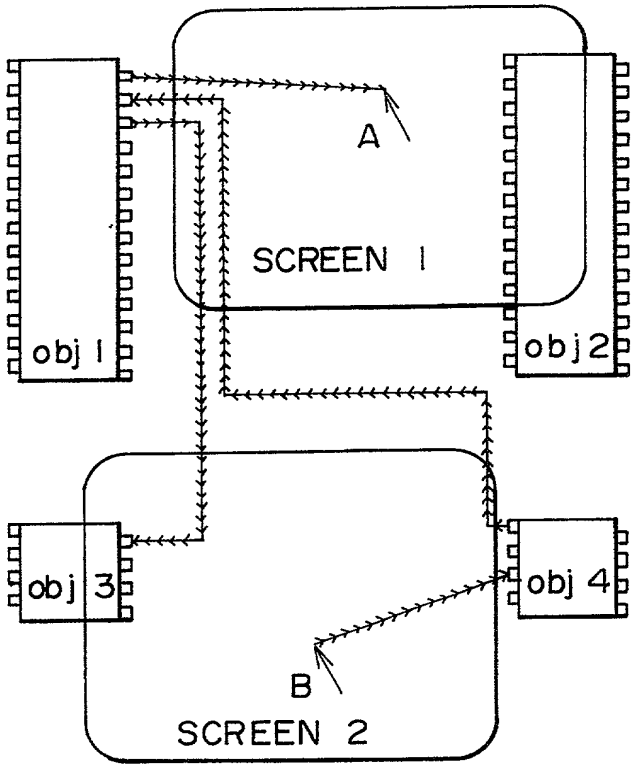


Fig.52(B)

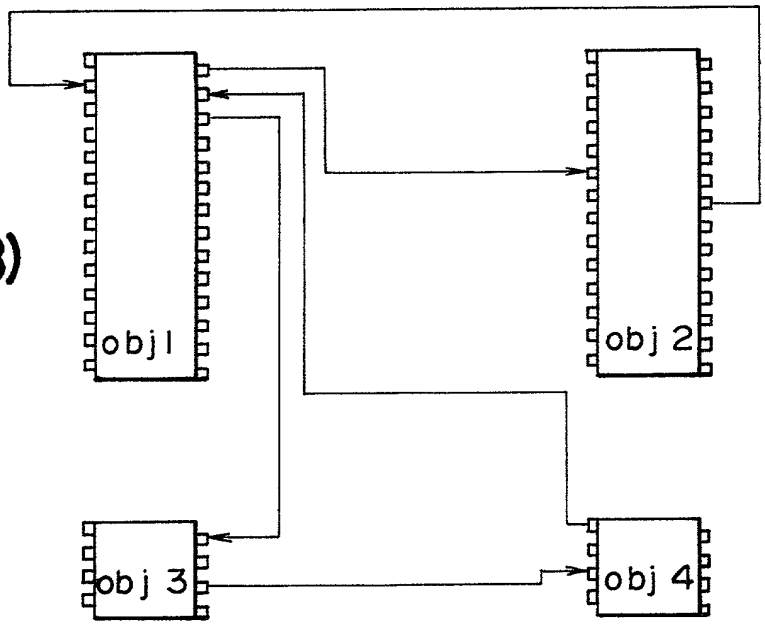


Fig.53

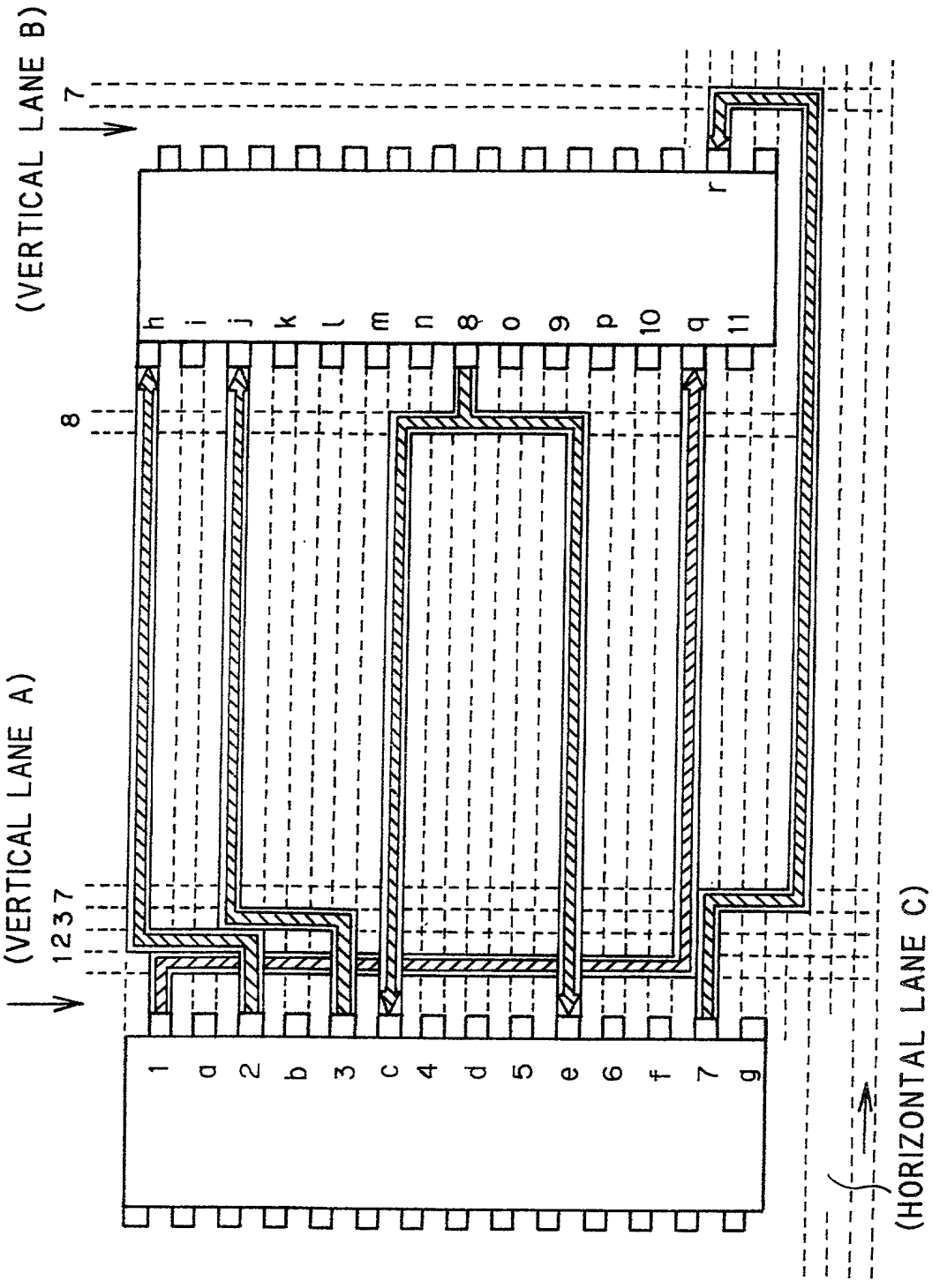

(VERTICAL LANE B)
7
8
(VERTICAL LANE A)
1237
(HORIZONTAL LANE C)
h i j k l m n 8 o 9 p 10 q 11 r
1 a 2 b 3 c 4 d 5 e 6 f 7 g

Fig. 54

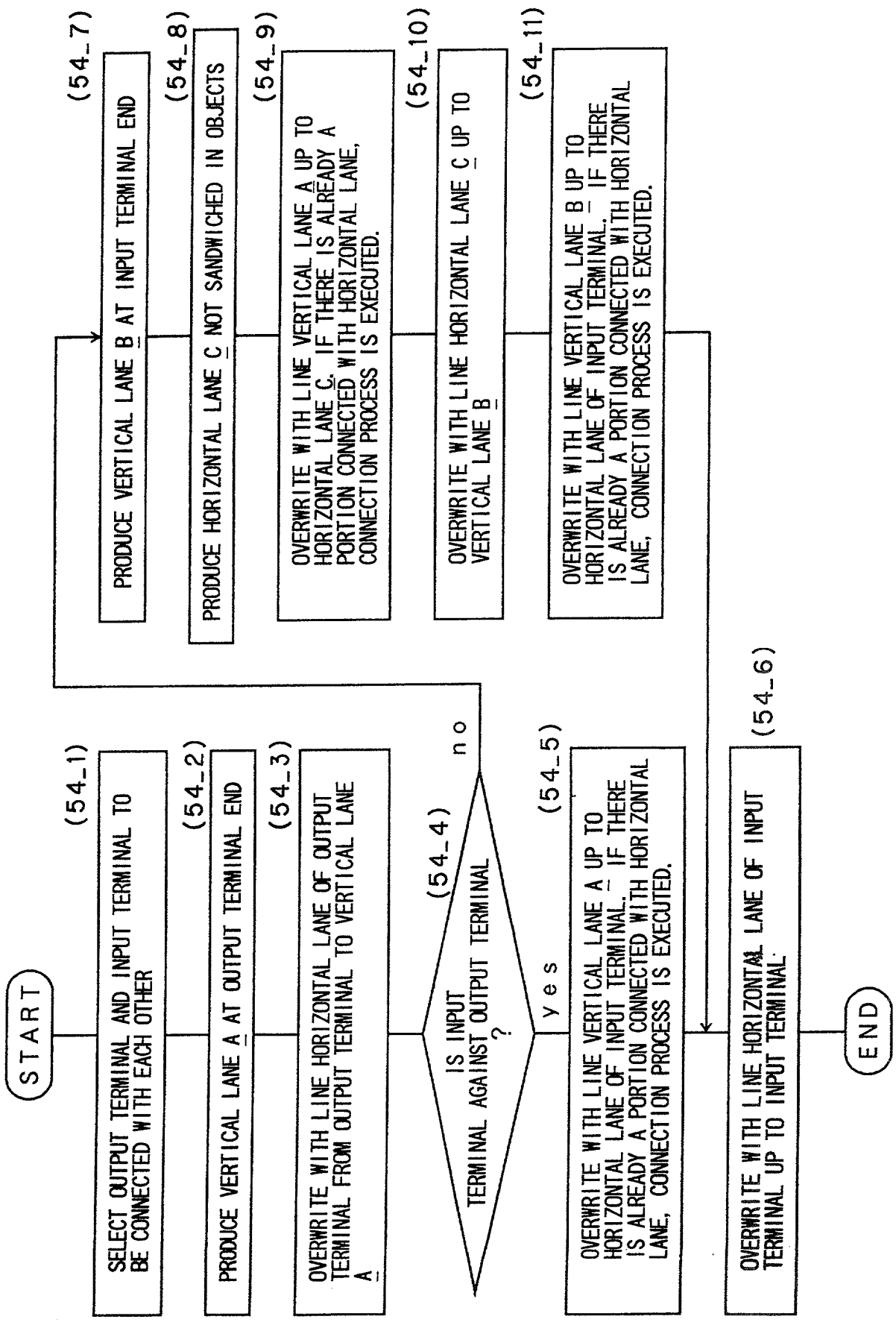

START
(54_1)
SELECT OUTPUT TERMINAL AND INPUT TERMINAL TO BE CONNECTED WITH EACH OTHER
(54_2)
PRODUCE VERTICAL LANE A AT OUTPUT TERMINAL END
(54_3)
OVERWRITE WITH LINE HORIZONTAL LANE OF OUTPUT TERMINAL FROM OUTPUT TERMINAL TO VERTICAL LANE A
(54_4)
IS INPUT TERMINAL AGAINST OUTPUT TERMINAL ?
no
yes
(54_5)
OVERWRITE WITH LINE VERTICAL LANE A UP TO HORIZONTAL LANE OF INPUT TERMINAL. IF THERE IS ALREADY A PORTION CONNECTED WITH HORIZONTAL LANE, CONNECTION PROCESS IS EXECUTED.
(54_6)
OVERWRITE WITH LINE HORIZONTAL LANE OF INPUT TERMINAL UP TO INPUT TERMINAL
END
(54_7)
PRODUCE VERTICAL LANE B AT INPUT TERMINAL END
(54_8)
PRODUCE HORIZONTAL LANE C NOT SANDWICHED IN OBJECTS
(54_9)
OVERWRITE WITH LINE VERTICAL LANE A UP TO HORIZONTAL LANE C. IF THERE IS ALREADY A PORTION CONNECTED WITH HORIZONTAL LANE, CONNECTION PROCESS IS EXECUTED.
(54_10)
OVERWRITE WITH LINE HORIZONTAL LANE C UP TO VERTICAL LANE B
(54_11)
OVERWRITE WITH LINE VERTICAL LANE B UP TO HORIZONTAL LANE OF INPUT TERMINAL. IF THERE IS ALREADY A PORTION CONNECTED WITH HORIZONTAL LANE, CONNECTION PROCESS IS EXECUTED.

Fig.55

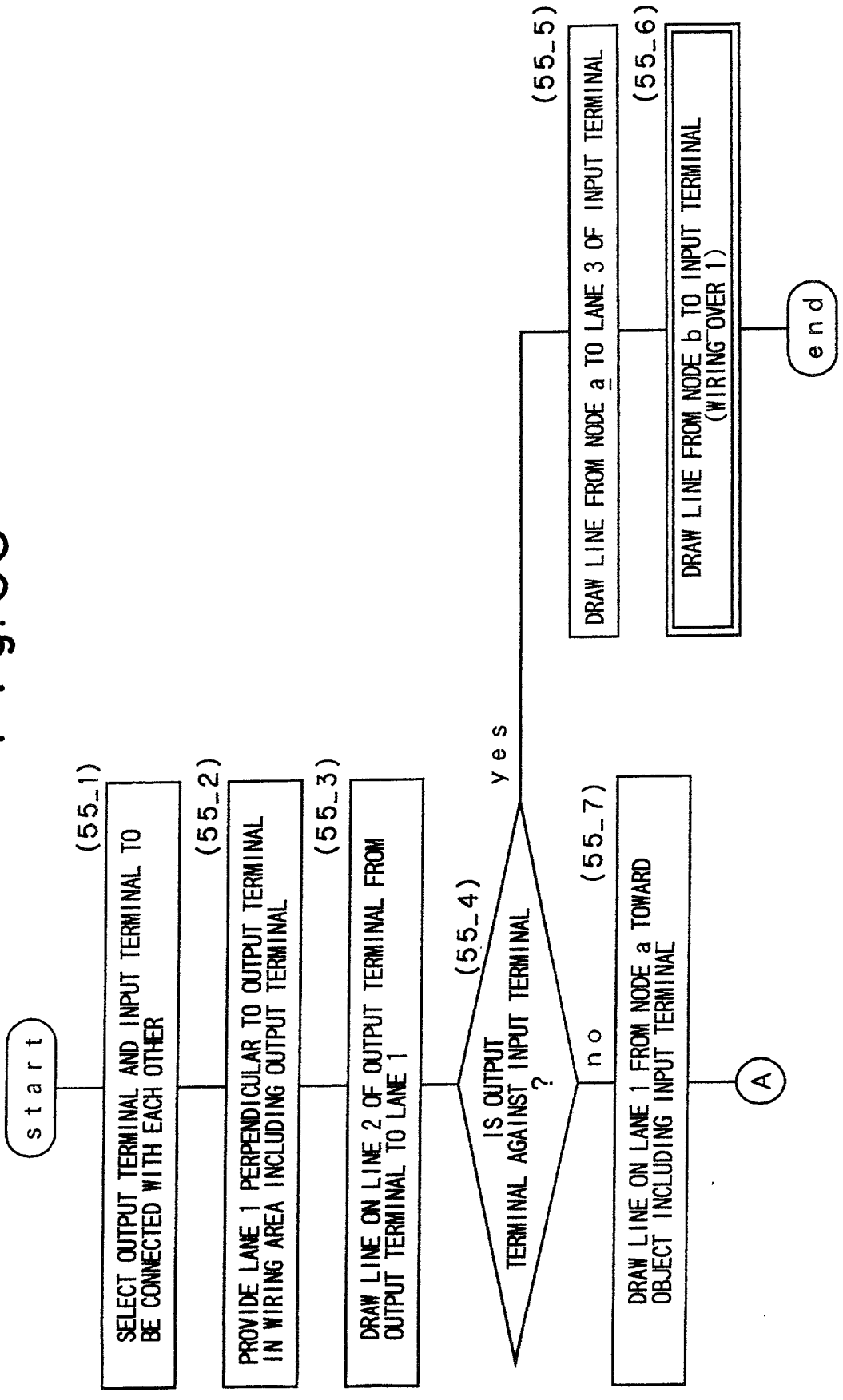

start
(55_1)
SELECT OUTPUT TERMINAL AND INPUT TERMINAL TO BE CONNECTED WITH EACH OTHER
(55_2)
PROVIDE LANE 1 PERPENDICULAR TO OUTPUT TERMINAL IN WIRING AREA INCLUDING OUTPUT TERMINAL
(55_3)
DRAW LINE ON LINE 2 OF OUTPUT TERMINAL FROM OUTPUT TERMINAL TO LANE 1
(55_4)
IS OUTPUT TERMINAL AGAINST INPUT TERMINAL ?
yes
(55_5)
DRAW LINE FROM NODE a TO LANE 3 OF INPUT TERMINAL
(55_6)
DRAW LINE FROM NODE b TO INPUT TERMINAL (WIRING OVER 1)
end
no
(55_7)
DRAW LINE ON LANE 1 FROM NODE a TOWARD OBJECT INCLUDING INPUT TERMINAL
A

Fig.56

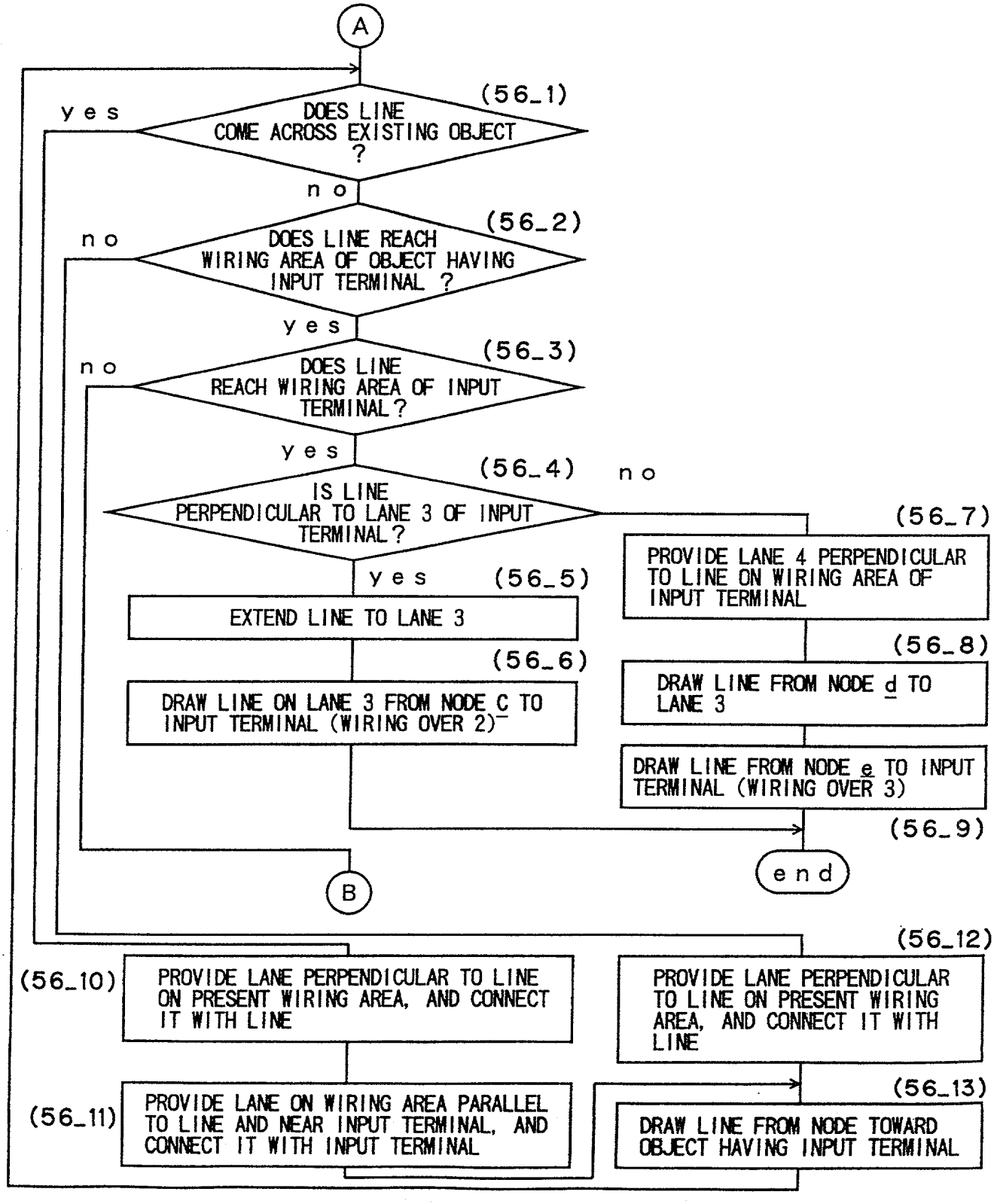

A
(56_1)
DOES LINE COME ACROSS EXISTING OBJECT ?
yes
no
(56_2)
DOES LINE REACH WIRING AREA OF OBJECT HAVING INPUT TERMINAL ?
no
yes
(56_3)
DOES LINE REACH WIRING AREA OF INPUT TERMINAL?
no
yes
(56_4)
IS LINE PERPENDICULAR TO LANE 3 OF INPUT TERMINAL?
no
yes
(56_5)
EXTEND LINE TO LANE 3
(56_6)
DRAW LINE ON LANE 3 FROM NODE C TO INPUT TERMINAL (WIRING OVER 2)
(56_7)
PROVIDE LANE 4 PERPENDICULAR TO LINE ON WIRING AREA OF INPUT TERMINAL
(56_8)
DRAW LINE FROM NODE d TO LANE 3
DRAW LINE FROM NODE e TO INPUT TERMINAL (WIRING OVER 3)
(56_9)
end
B
(56_10)
PROVIDE LANE PERPENDICULAR TO LINE ON PRESENT WIRING AREA, AND CONNECT IT WITH LINE
(56_11)
PROVIDE LANE ON WIRING AREA PARALLEL TO LINE AND NEAR INPUT TERMINAL, AND CONNECT IT WITH INPUT TERMINAL
(56_12)
PROVIDE LANE PERPENDICULAR TO LINE ON PRESENT WIRING AREA, AND CONNECT IT WITH LINE
(56_13)
DRAW LINE FROM NODE TOWARD OBJECT HAVING INPUT TERMINAL

Fig. 57

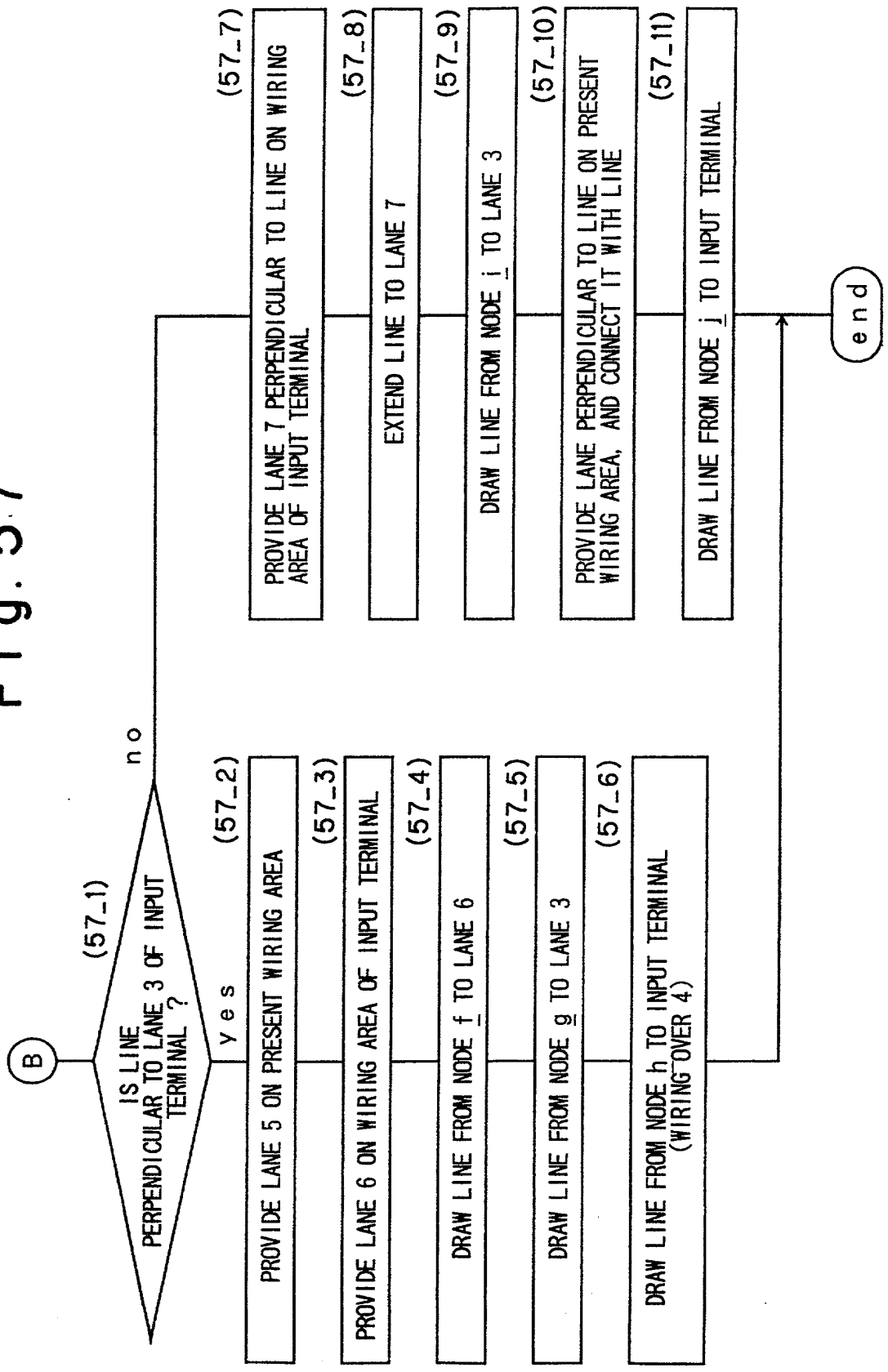

B
(57_1)
IS LINE PERPENDICULAR TO LANE 3 OF INPUT TERMINAL ?
yes
no
(57_2)
PROVIDE LANE 5 ON PRESENT WIRING AREA
(57_3)
PROVIDE LANE 6 ON WIRING AREA OF INPUT TERMINAL
(57_4)
DRAW LINE FROM NODE f TO LANE 6
(57_5)
DRAW LINE FROM NODE g TO LANE 3
(57_6)
DRAW LINE FROM NODE h TO INPUT TERMINAL (WIRING OVER 4)
(57_7)
PROVIDE LANE 7 PERPENDICULAR TO LINE ON WIRING AREA OF INPUT TERMINAL
(57_8)
EXTEND LINE TO LANE 7
(57_9)
DRAW LINE FROM NODE i TO LANE 3
(57_10)
PROVIDE LANE PERPENDICULAR TO LINE ON PRESENT WIRING AREA, AND CONNECT IT WITH LINE
(57_11)
DRAW LINE FROM NODE j TO INPUT TERMINAL
end

Fig. 58

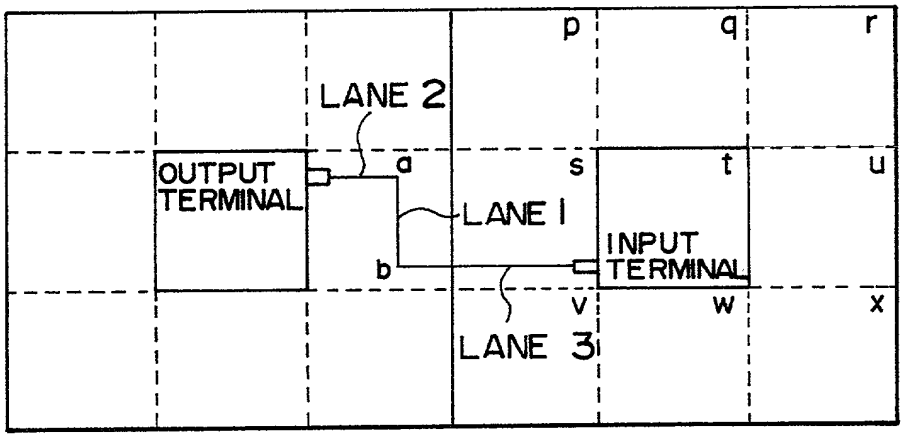

p
q
r
LANE 2
OUTPUT TERMINAL
a
s
t
u
LANE 1
INPUT TERMINAL
b
v
w
x
LANE 3

Fig. 59

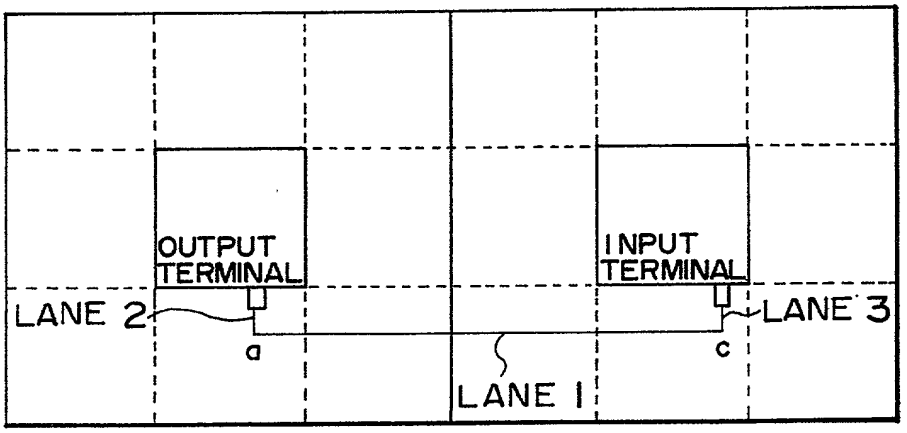

OUTPUT TERMINAL
INPUT TERMINAL
LANE 2
LANE 3
a
c
LANE 1

Fig. 60

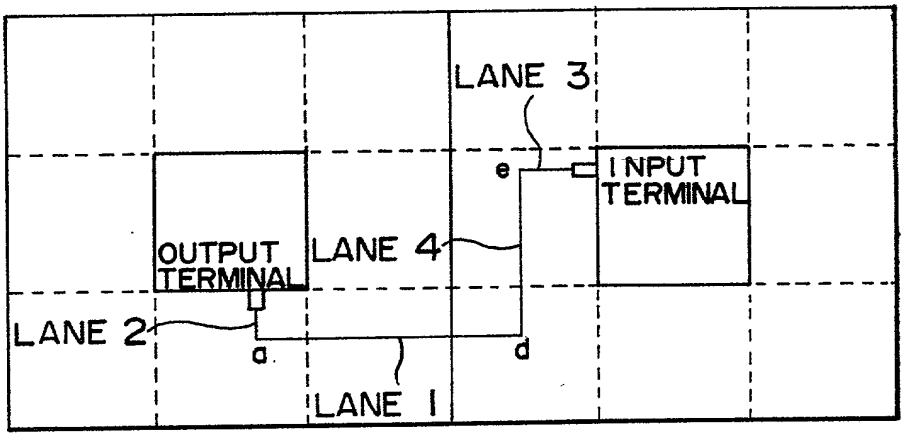

LANE 3
e
INPUT TERMINAL
OUTPUT TERMINAL
LANE 4
LANE 2
a
d
LANE 1

Fig. 61

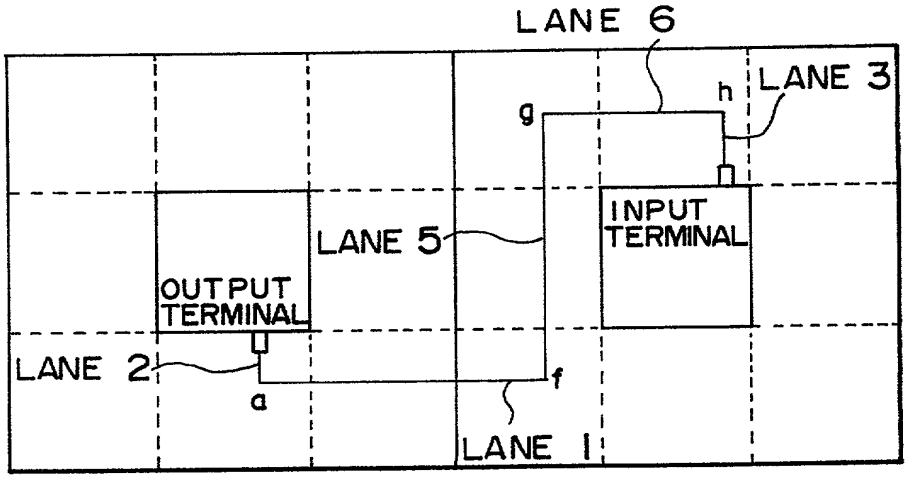

LANE 6
h
LANE 3
g
INPUT TERMINAL
LANE 5
OUTPUT TERMINAL
LANE 2
a
f
LANE 1

Fig. 62

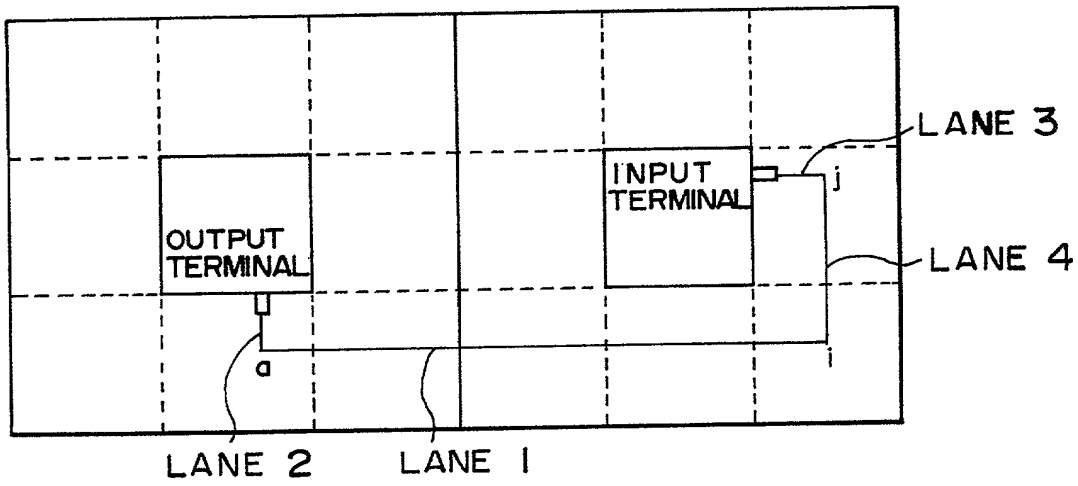

LANE 3
INPUT TERMINAL
j
OUTPUT TERMINAL
LANE 4
a
i
LANE 2
LANE 1

Fig.63(A)

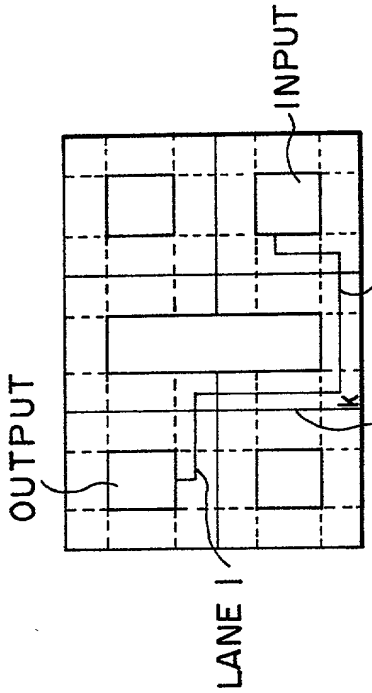


Fig.63(B)

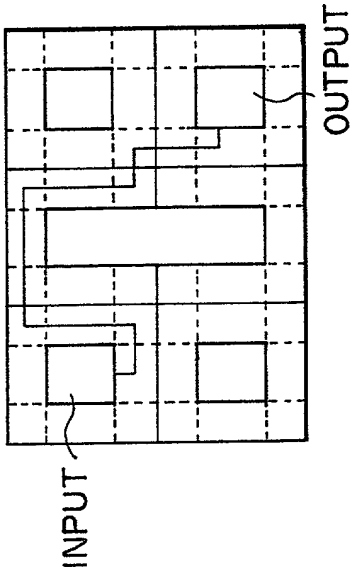


Fig.63(C)

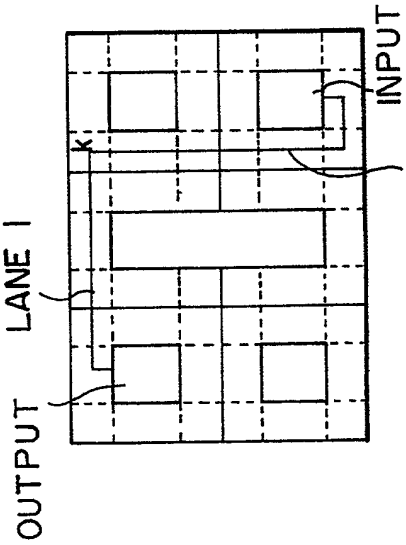


Fig.63(D)

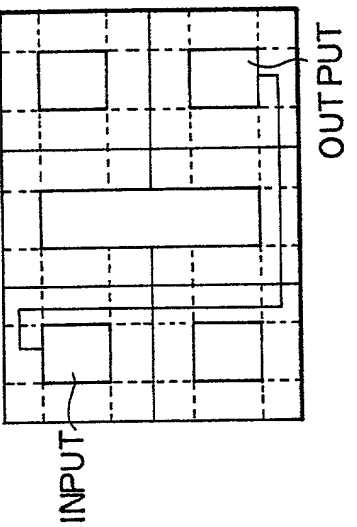


Fig. 64

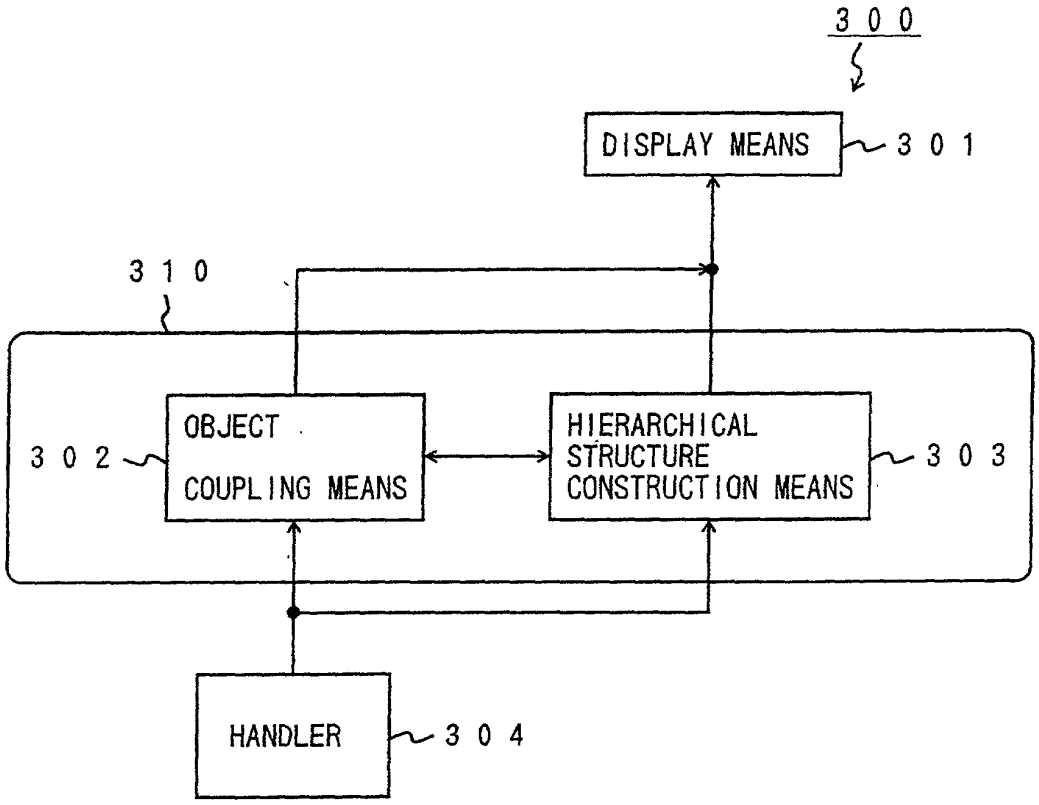

300
DISPLAY MEANS
301
310
302
OBJECT
COUPLING MEANS
HIERARCHICAL
STRUCTURE
CONSTRUCTION MEANS
303
HANDLER
304

Fig.65

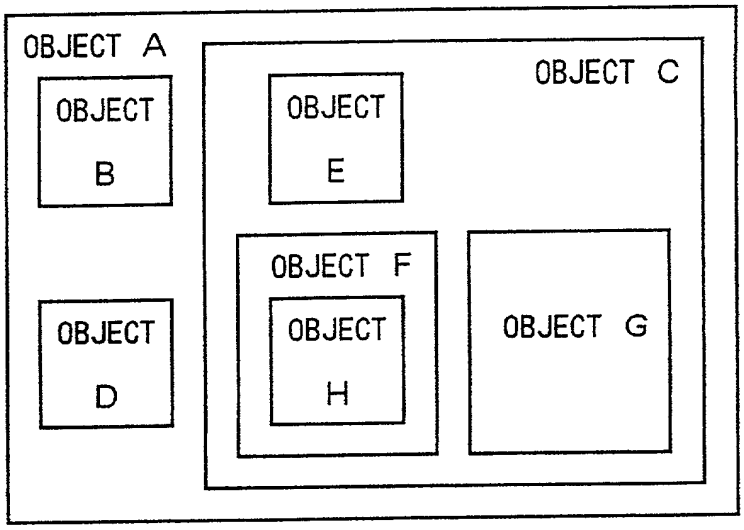

OBJECT A
OBJECT B
OBJECT C
OBJECT E
OBJECT F
OBJECT H
OBJECT G
OBJECT D

Fig.66

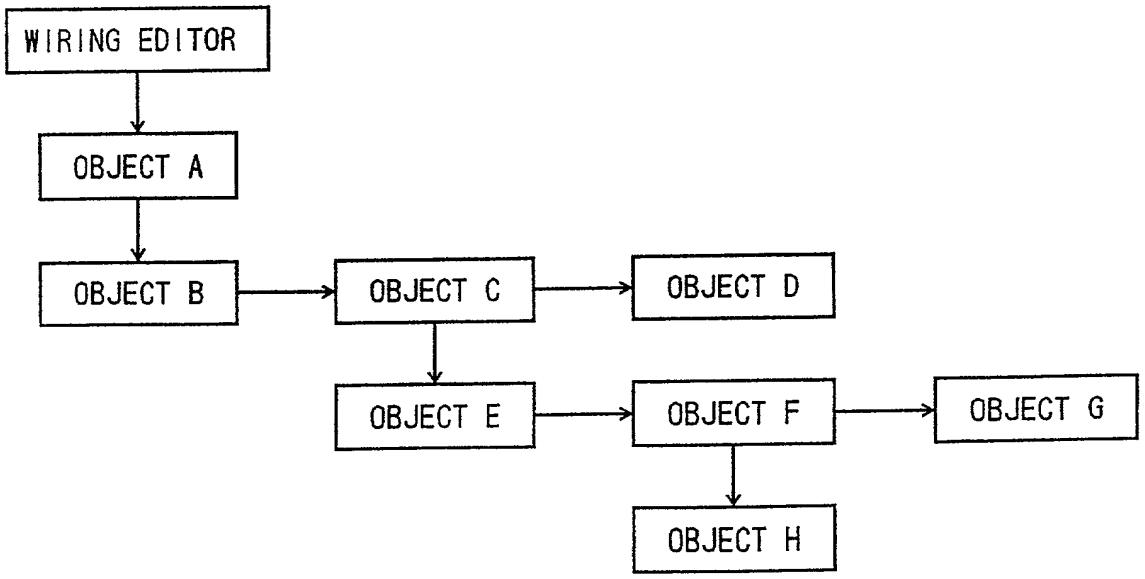

WIRING EDITOR
OBJECT A
OBJECT B
OBJECT C
OBJECT D
OBJECT E
OBJECT F
OBJECT G
OBJECT H

Fig. 67

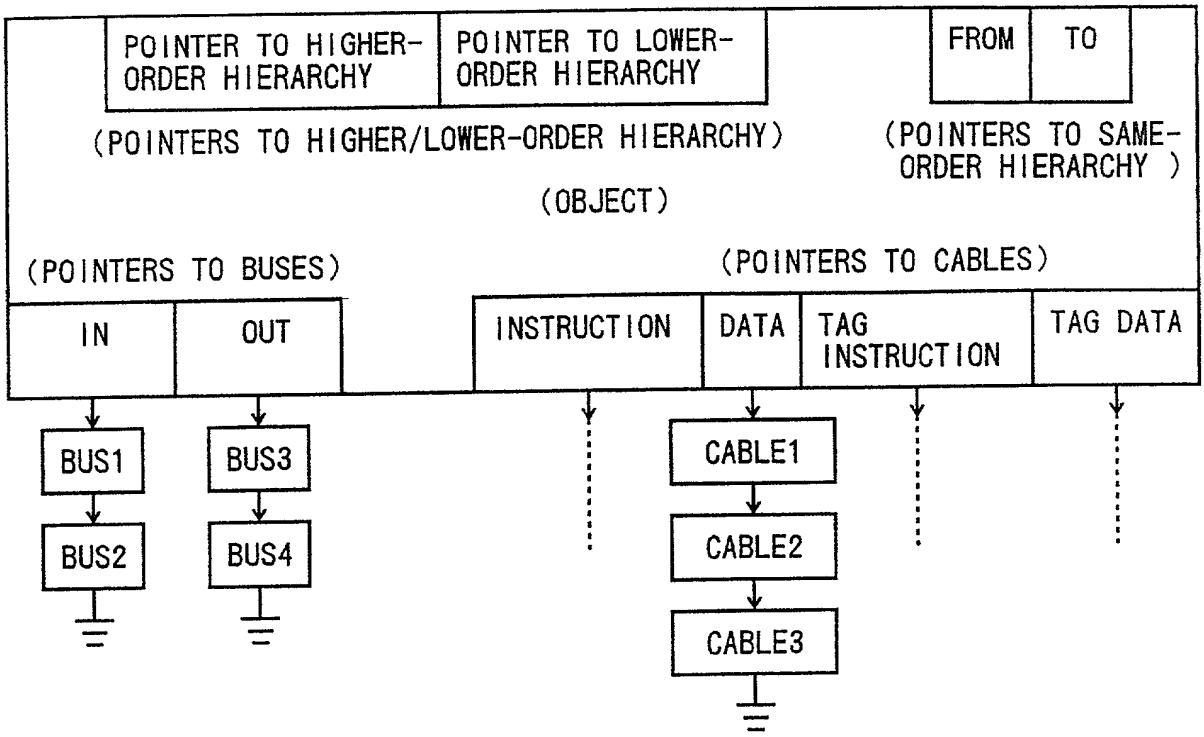


Fig. 68

(BUS)

POINTER TO SUBSTANTIAL OBJECT
POINTER TO BUS OF SUBSTANTIAL OBJECT
POINTER TO NEXT BUS
OTHER DATA

Fig. 69

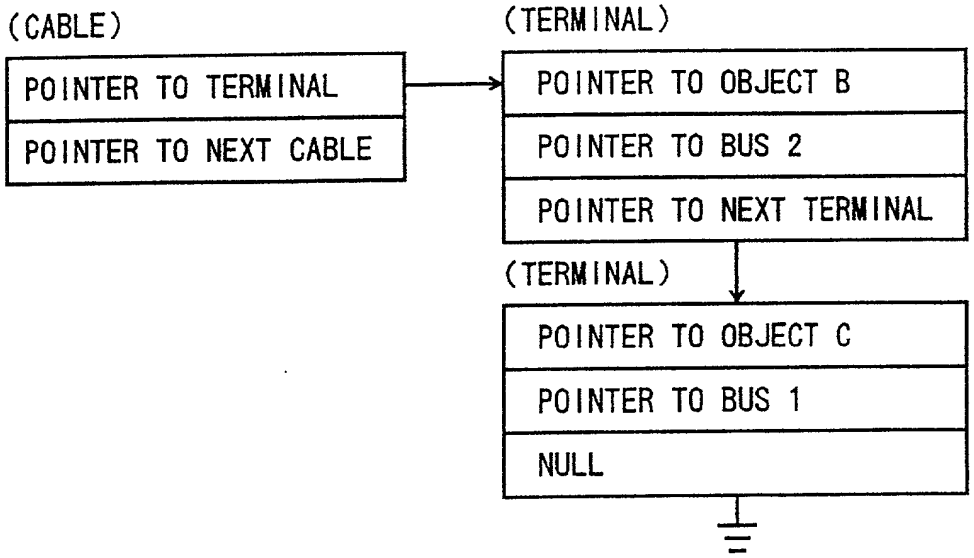

(CABLE)
POINTER TO TERMINAL
POINTER TO NEXT CABLE
(TERMINAL)
POINTER TO OBJECT B
POINTER TO BUS 2
POINTER TO NEXT TERMINAL
(TERMINAL)
POINTER TO OBJECT C
POINTER TO BUS 1
NULL

Fig. 70

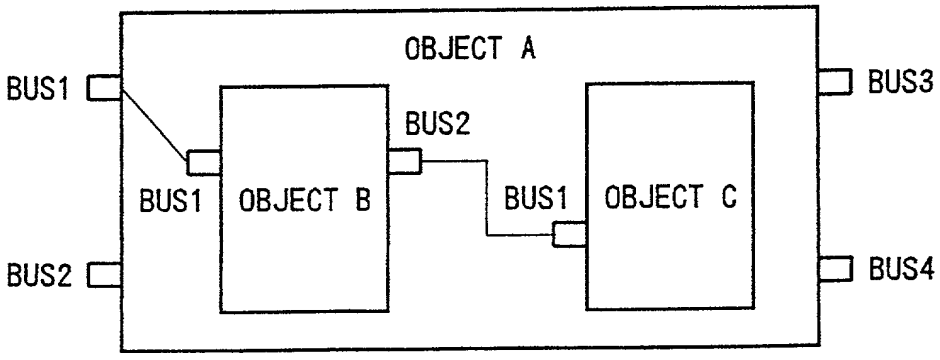

OBJECT A
BUS1
BUS2
BUS3
BUS4
BUS1
OBJECT B
BUS2
BUS1
OBJECT C

Fig. 71

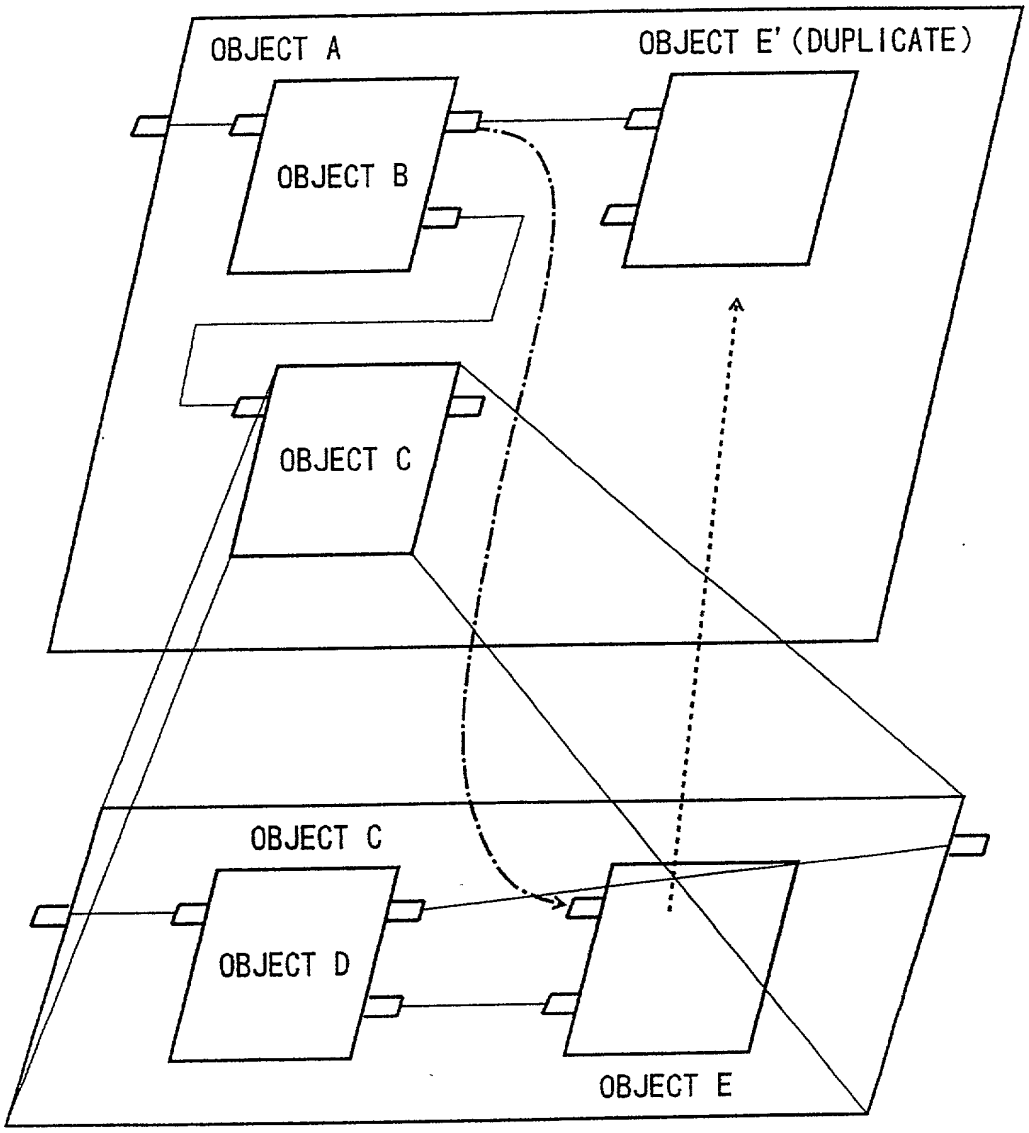

OBJECT A
OBJECT E'(DUPLICATE)
OBJECT B
OBJECT C
OBJECT C
OBJECT D
OBJECT E

Fig. 72

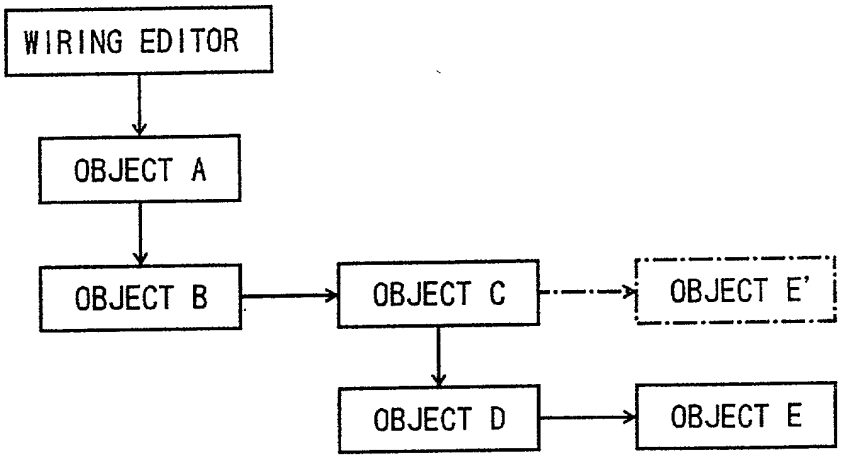

WIRING EDITOR
OBJECT A
OBJECT B
OBJECT C
OBJECT E'
OBJECT D
OBJECT E

Fig. 73

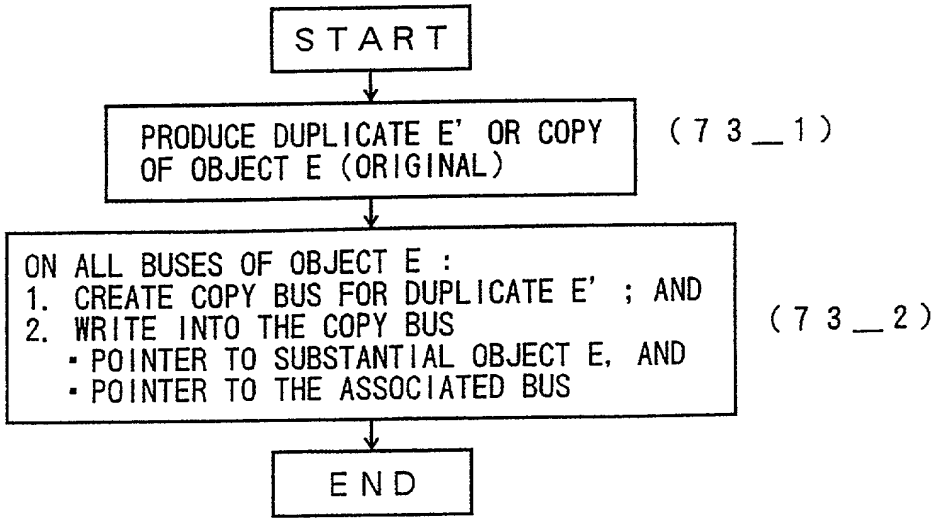

START
PRODUCE DUPLICATE E' OR COPY OF OBJECT E (ORIGINAL)
(73_1)
ON ALL BUSES OF OBJECT E :
1. CREATE COPY BUS FOR DUPLICATE E' ; AND
2. WRITE INTO THE COPY BUS
• POINTER TO SUBSTANTIAL OBJECT E, AND
• POINTER TO THE ASSOCIATED BUS
(73_2)
END

Fig. 74

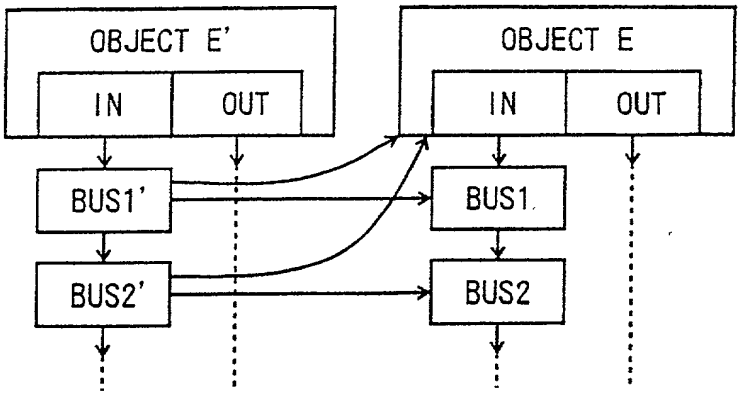

OBJECT E'
IN
OUT
OBJECT E
IN
OUT
BUS1'
BUS2'
BUS1
BUS2

Fig. 75

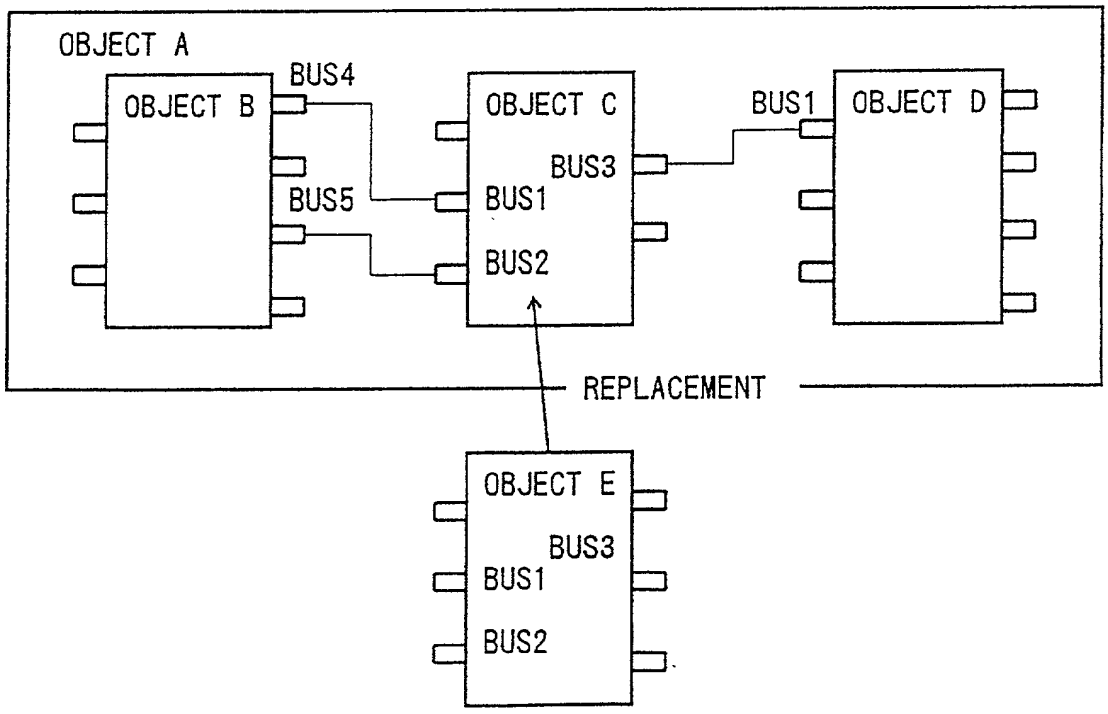

OBJECT A
OBJECT B
BUS4
BUS5
OBJECT C
BUS3
BUS1
BUS2
BUS1
OBJECT D
REPLACEMENT
OBJECT E
BUS3
BUS1
BUS2

Fig. 76

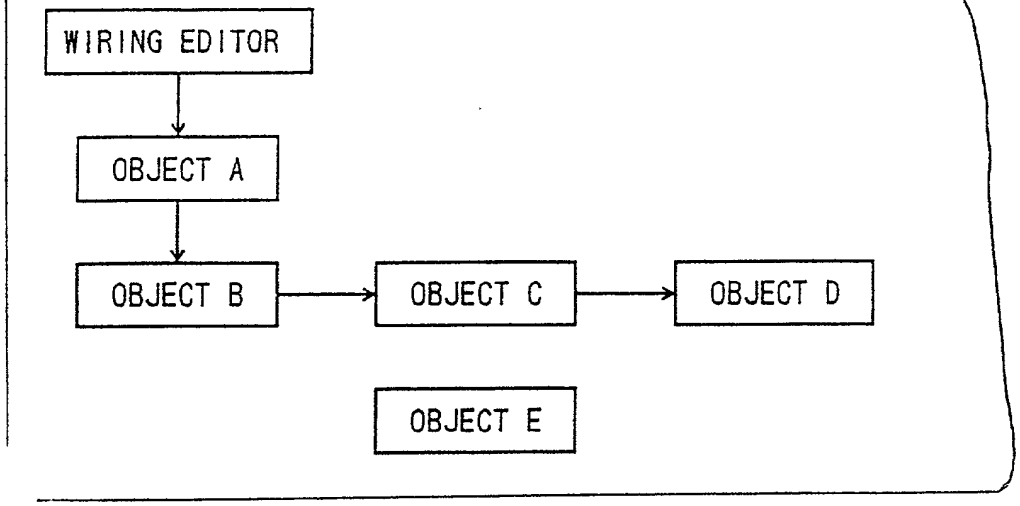

WIRING EDITOR
OBJECT A
OBJECT B
OBJECT C
OBJECT D
OBJECT E

Fig. 77

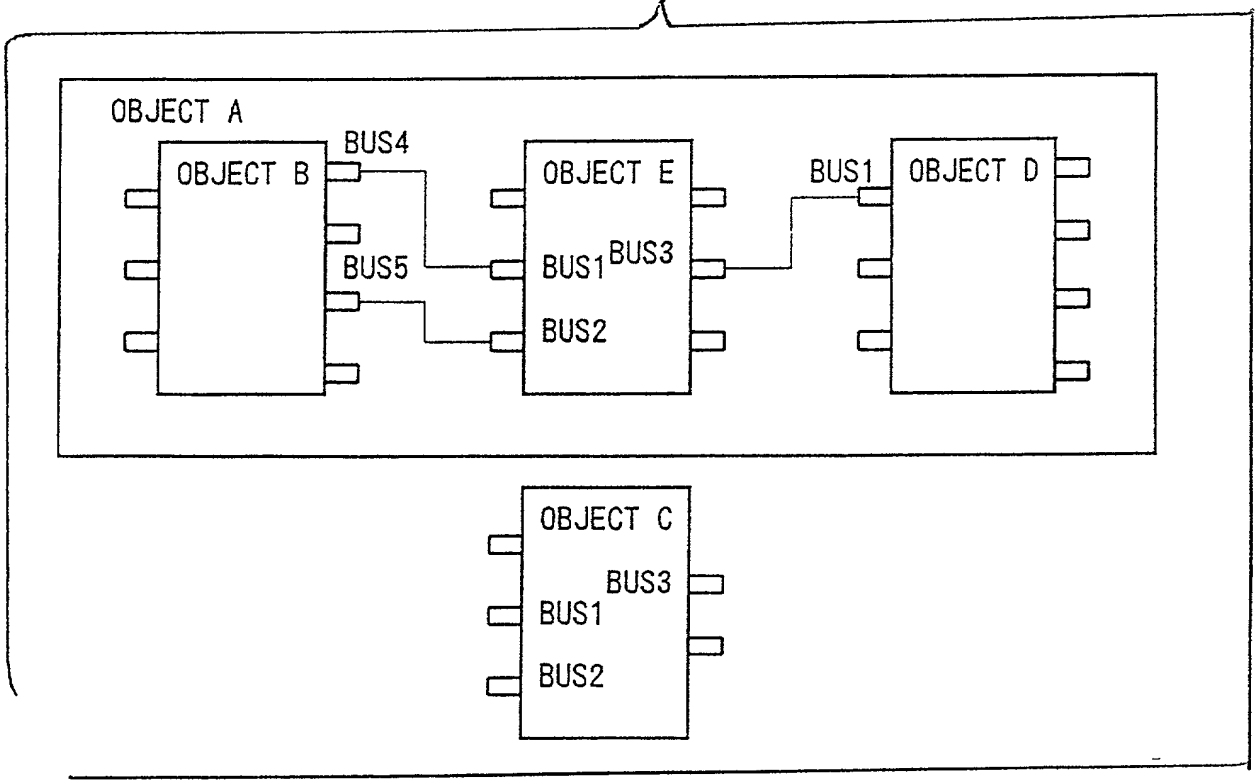

OBJECT A
OBJECT B
BUS4
BUS5
OBJECT E
BUS1
BUS3
BUS2
BUS1
OBJECT D
OBJECT C
BUS3
BUS1
BUS2

Fig. 78

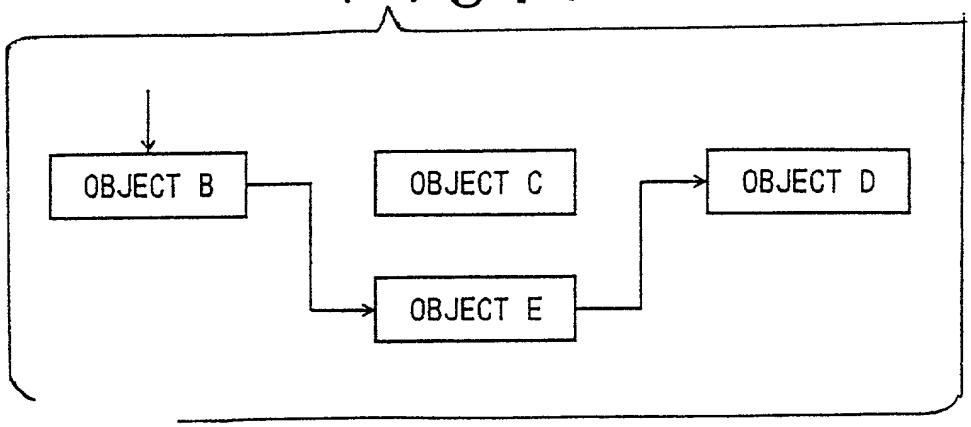

OBJECT B
OBJECT C
OBJECT D
OBJECT E

Fig. 79

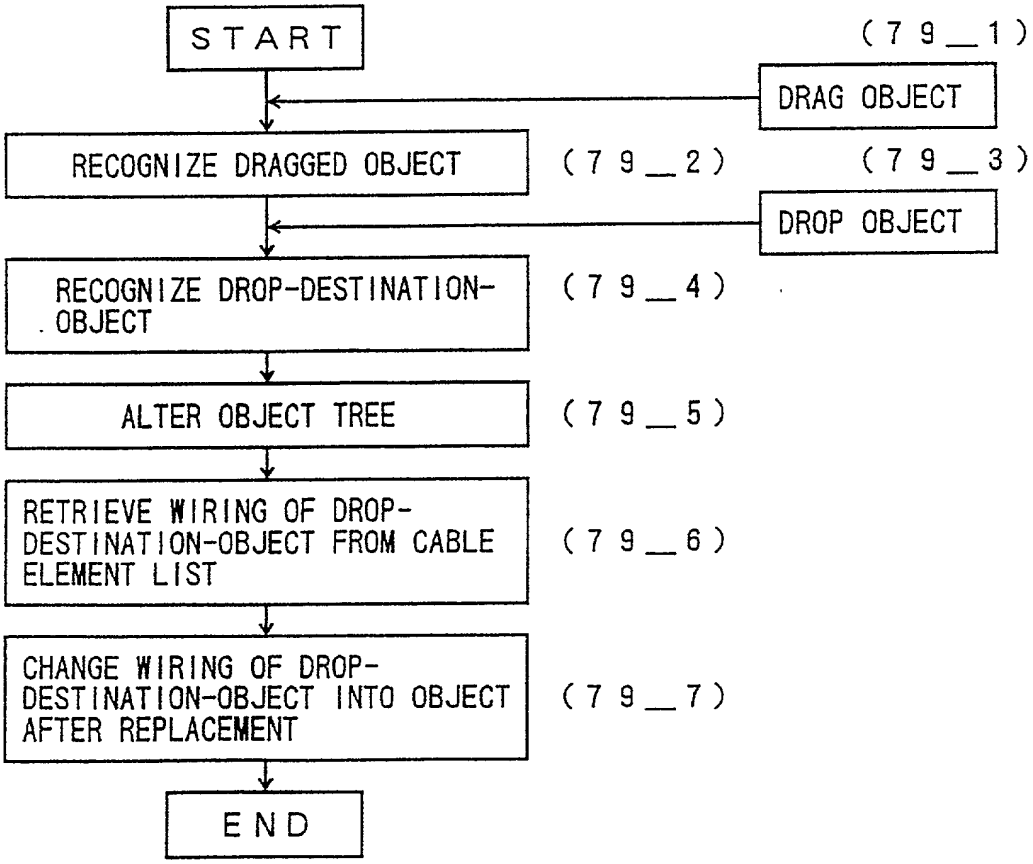

START
(79_1)
DRAG OBJECT
RECOGNIZE DRAGGED OBJECT
(79_2)
(79_3)
DROP OBJECT
RECOGNIZE DROP-DESTINATION-OBJECT
(79_4)
ALTER OBJECT TREE
(79_5)
RETRIEVE WIRING OF DROP-DESTINATION-OBJECT FROM CABLE ELEMENT LIST
(79_6)
CHANGE WIRING OF DROP-DESTINATION-OBJECT INTO OBJECT AFTER REPLACEMENT
(79_7)
END

Fig. 80

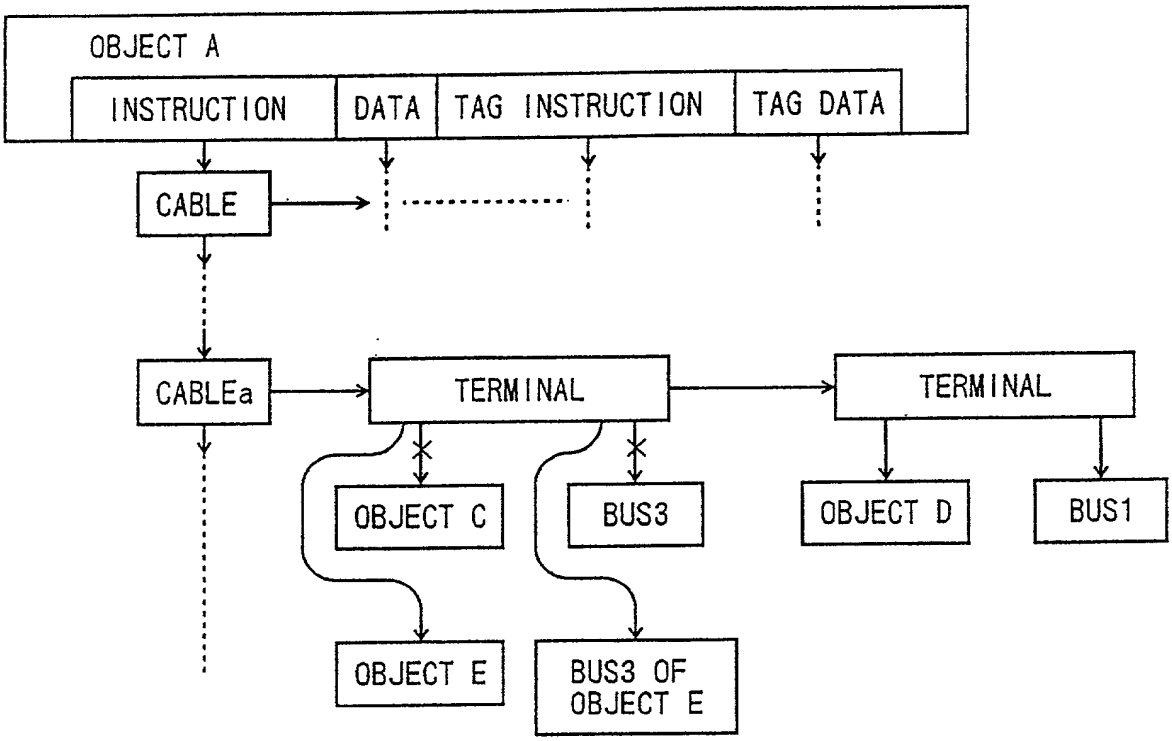

OBJECT A
INSTRUCTION
DATA
TAG INSTRUCTION
TAG DATA
CABLE
CABLEa
TERMINAL
TERMINAL
OBJECT C
BUS3
OBJECT D
BUS1
OBJECT E
BUS3 OF OBJECT E

Fig.81

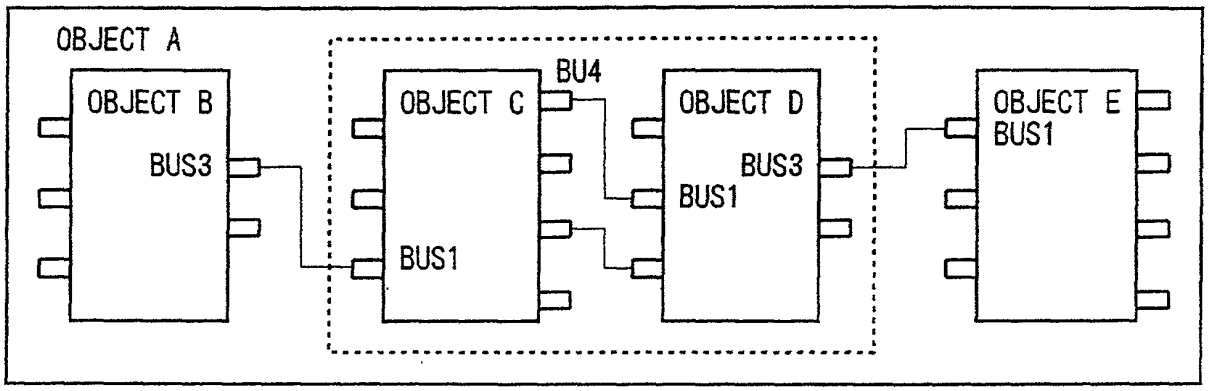

OBJECT A
OBJECT B
BUS3
OBJECT C
BU4
BUS1
OBJECT D
BUS3
BUS1
OBJECT E
BUS1

Fig. 82

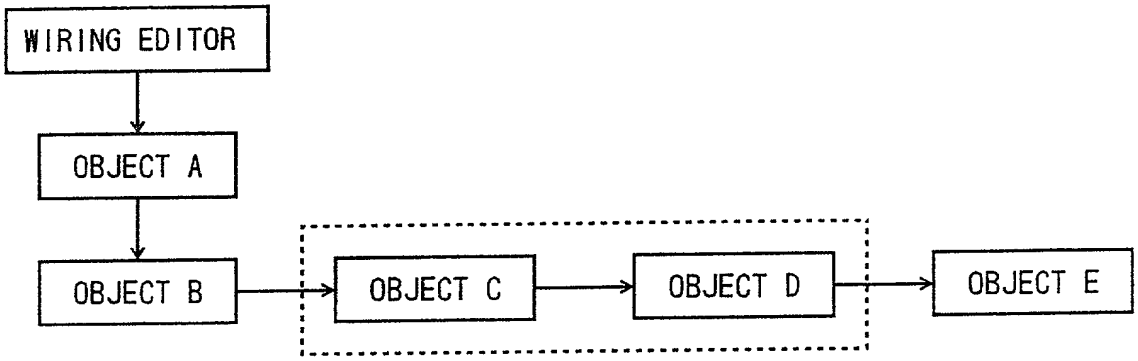


Fig. 83

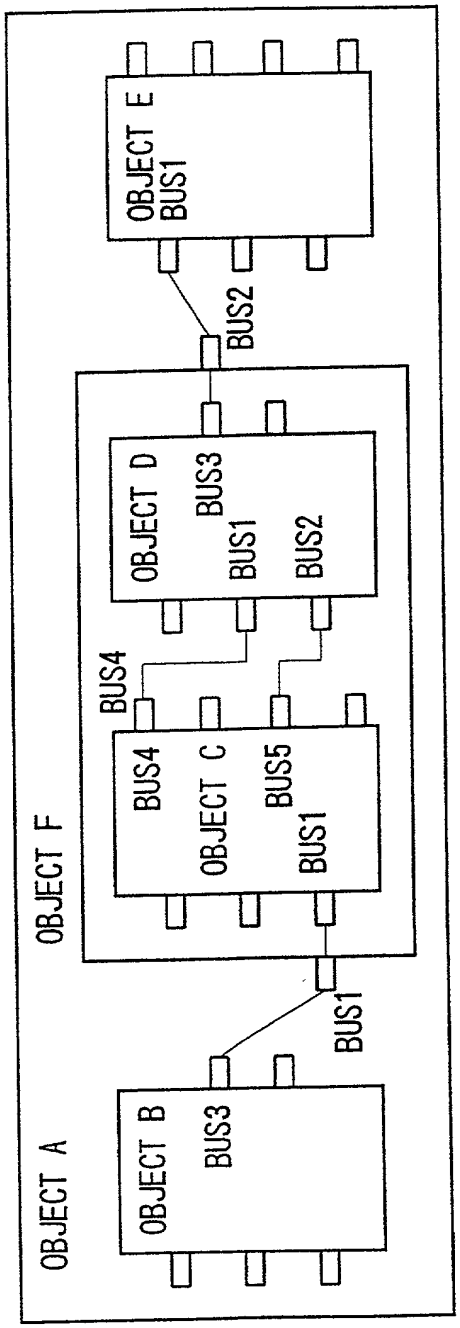

OBJECT A
OBJECT B
BUS3
BUS1
OBJECT F
BUS4
OBJECT C
BUS4
BUS5
BUS1
OBJECT D
BUS3
BUS1
BUS2
BUS2
OBJECT E
BUS1

Fig. 84

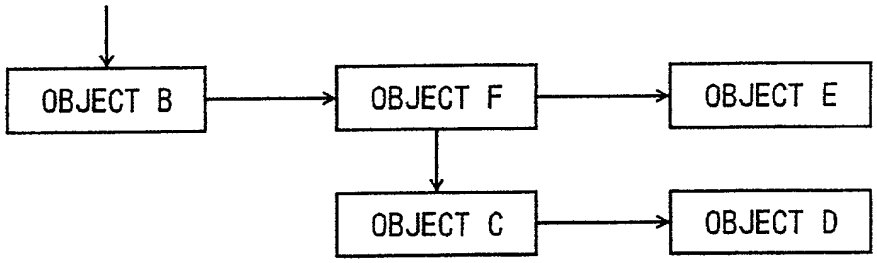


Fig. 85

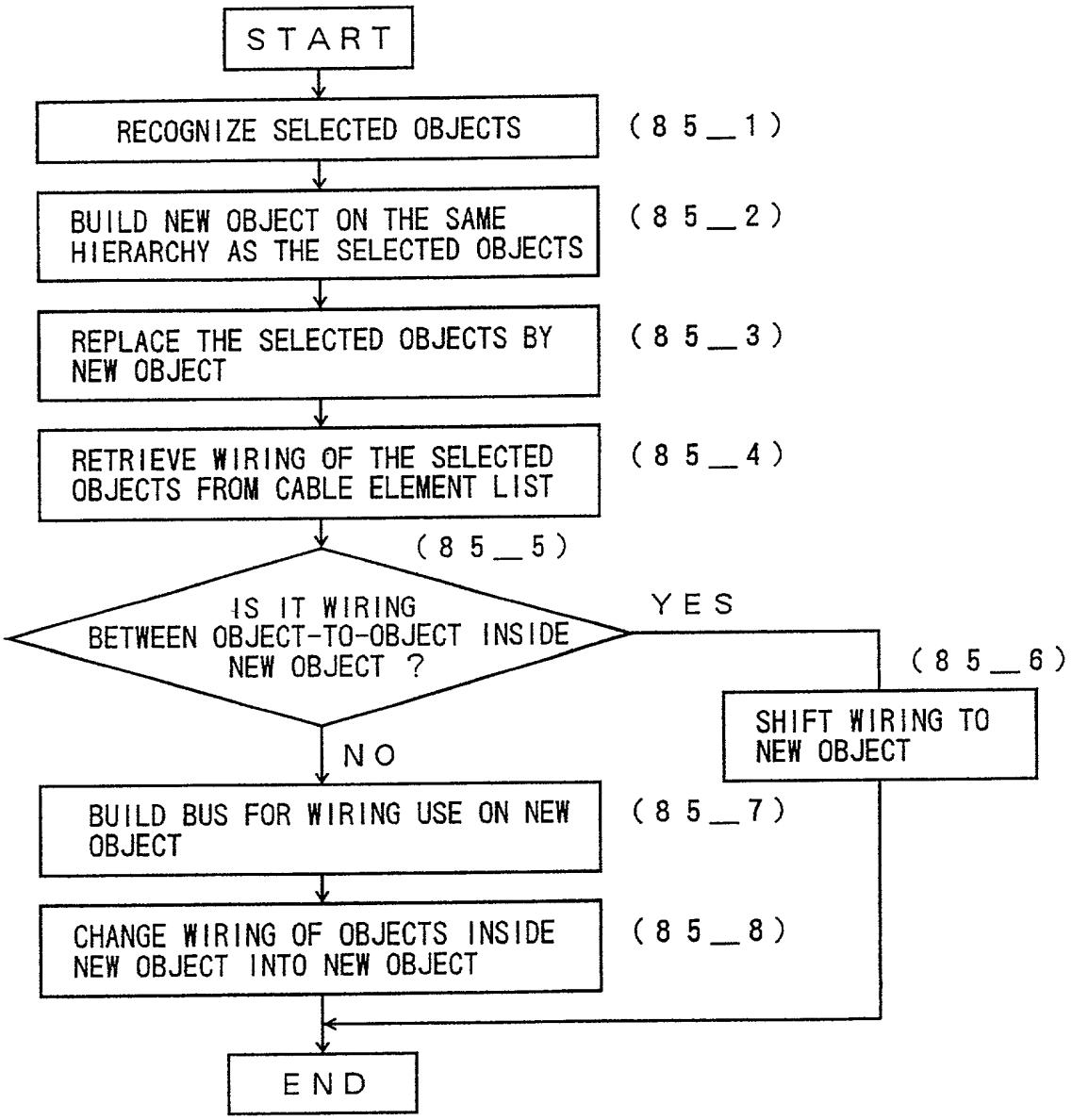

START
RECOGNIZE SELECTED OBJECTS
(85_1)
BUILD NEW OBJECT ON THE SAME HIERARCHY AS THE SELECTED OBJECTS
(85_2)
REPLACE THE SELECTED OBJECTS BY NEW OBJECT
(85_3)
RETRIEVE WIRING OF THE SELECTED OBJECTS FROM CABLE ELEMENT LIST
(85_4)
(85_5)
IS IT WIRING BETWEEN OBJECT-TO-OBJECT INSIDE NEW OBJECT ?
YES
(85_6)
SHIFT WIRING TO NEW OBJECT
NO
BUILD BUS FOR WIRING USE ON NEW OBJECT
(85_7)
CHANGE WIRING OF OBJECTS INSIDE NEW OBJECT INTO NEW OBJECT
(85_8)
END

Fig.86

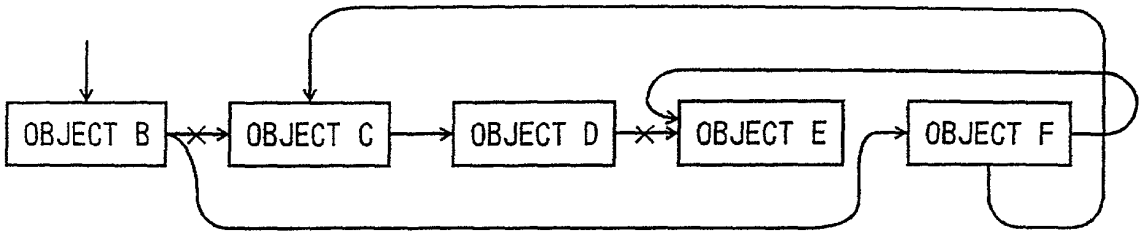

OBJECT B
OBJECT C
OBJECT D
OBJECT E
OBJECT F

Fig.87

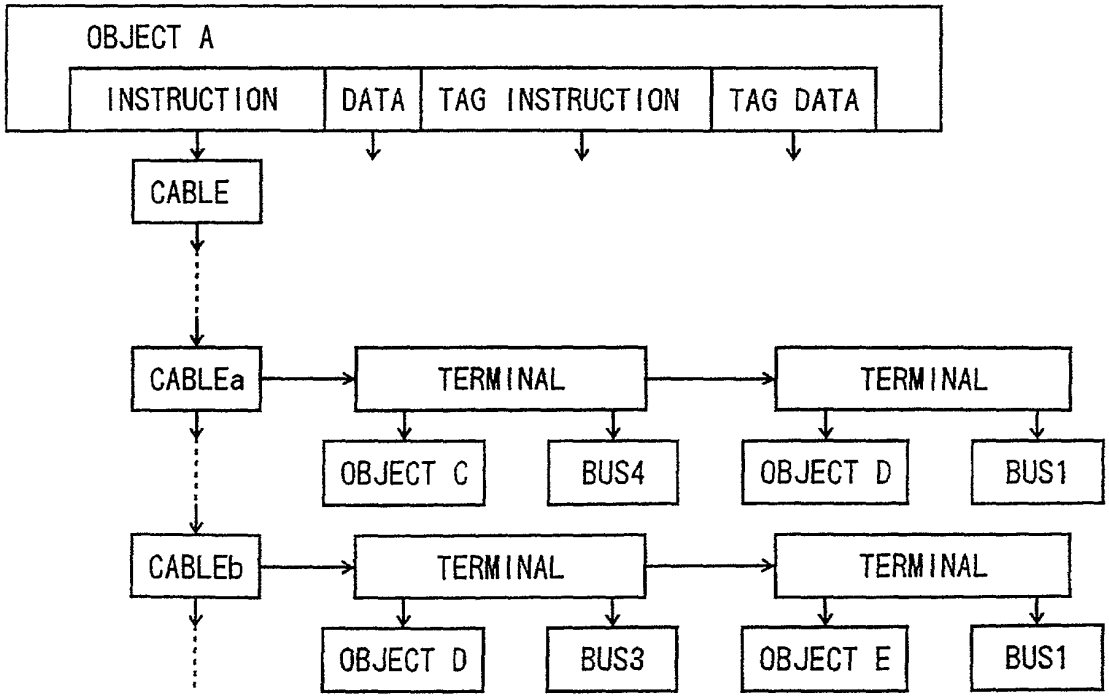

OBJECT A
INSTRUCTION
DATA
TAG INSTRUCTION
TAG DATA
CABLE
CABLEa
TERMINAL
TERMINAL
OBJECT C
BUS4
OBJECT D
BUS1
CABLEb
TERMINAL
TERMINAL
OBJECT D
BUS3
OBJECT E
BUS1

Fig.88

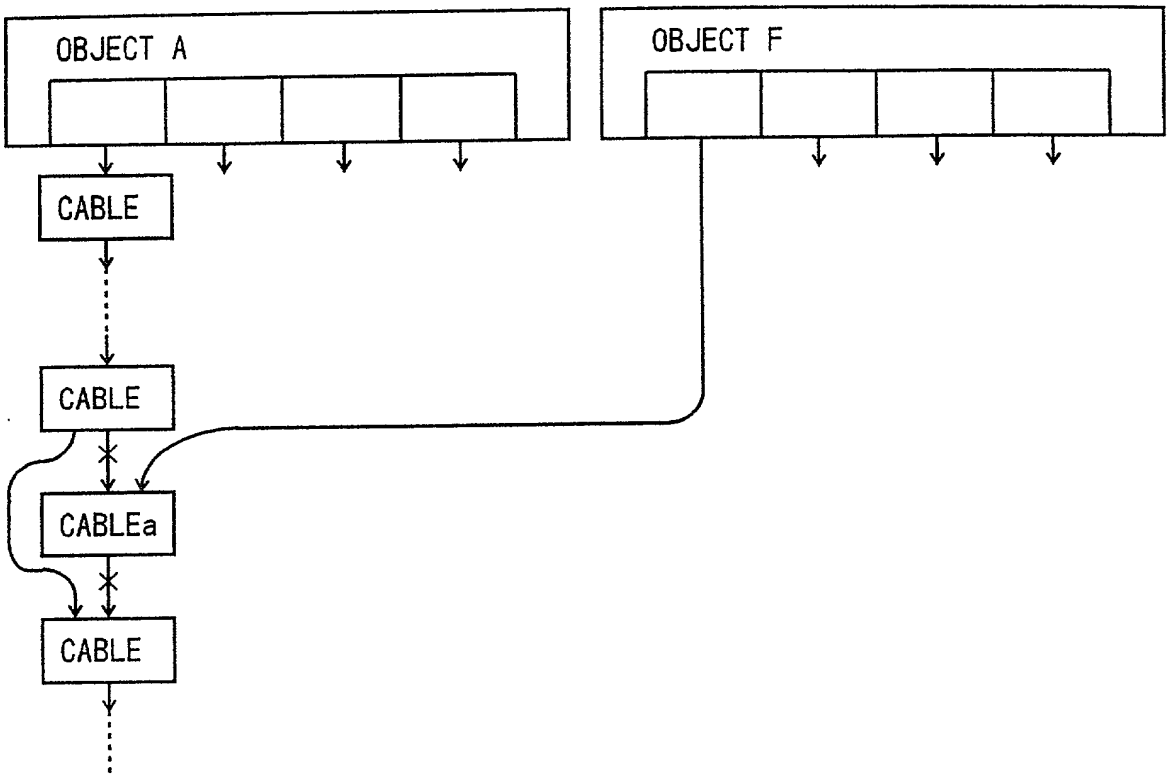

OBJECT A
OBJECT F
CABLE
CABLE
CABLEa
CABLE

Fig.89

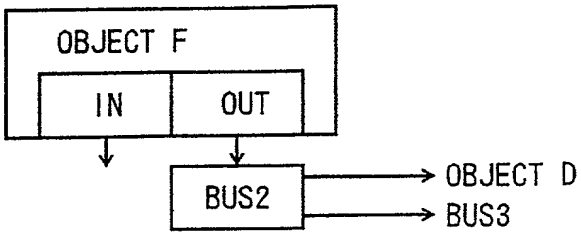

OBJECT F
IN
OUT
BUS2
OBJECT D
BUS3

Fig.90

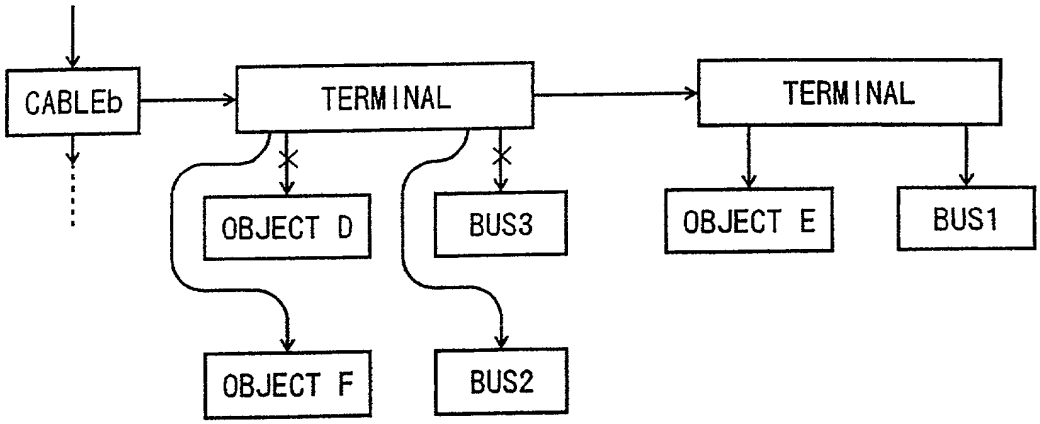

CABLEb
TERMINAL
TERMINAL
OBJECT D
BUS3
OBJECT E
BUS1
OBJECT F
BUS2

Fig.91

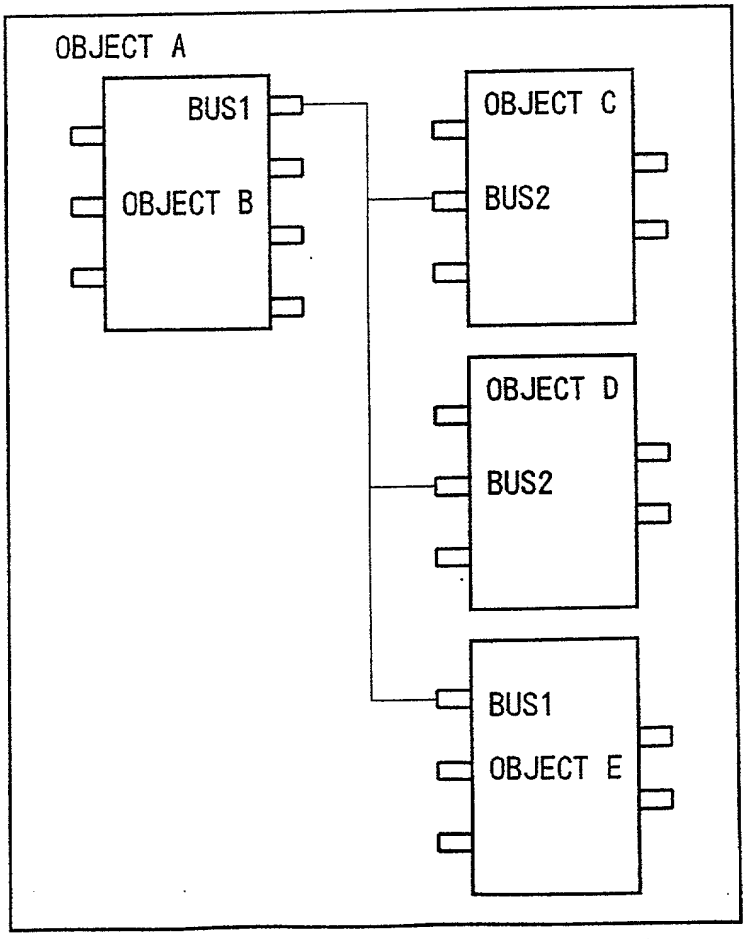

OBJECT A
BUS1
OBJECT B
OBJECT C
BUS2
OBJECT D
BUS2
BUS1
OBJECT E

Fig.92

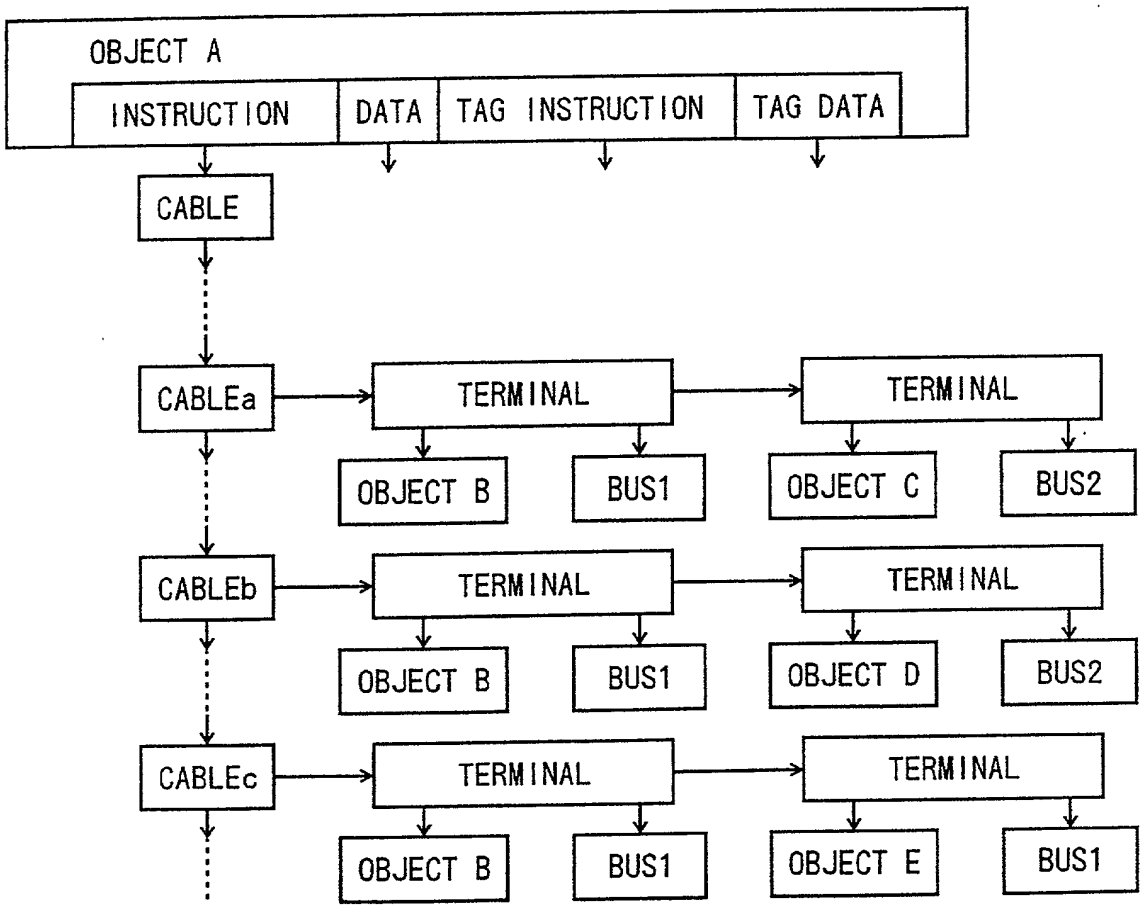

OBJECT A
INSTRUCTION
DATA
TAG INSTRUCTION
TAG DATA
CABLE
CABLEa
TERMINAL
TERMINAL
OBJECT B
BUS1
OBJECT C
BUS2
CABLEb
TERMINAL
TERMINAL
OBJECT B
BUS1
OBJECT D
BUS2
CABLEc
TERMINAL
TERMINAL
OBJECT B
BUS1
OBJECT E
BUS1

Fig. 93

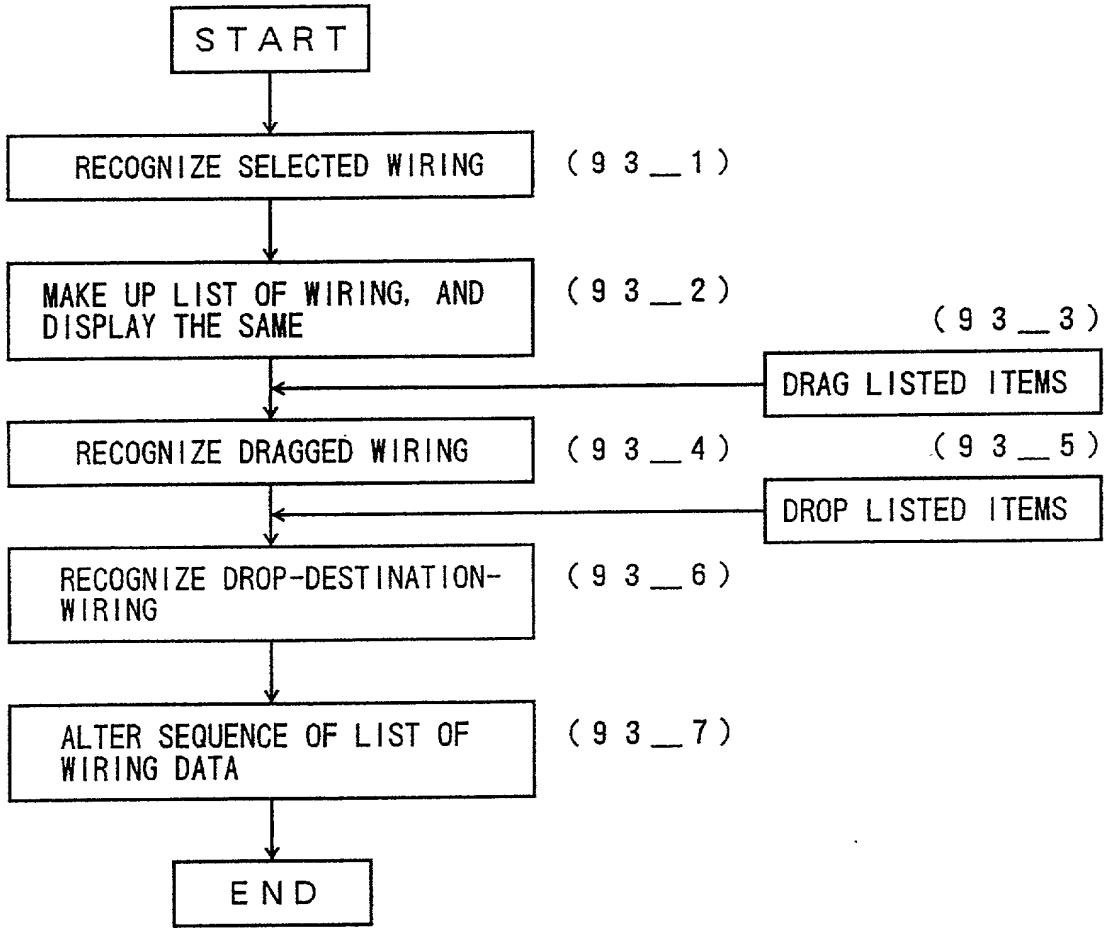

START
RECOGNIZE SELECTED WIRING
(93_1)
MAKE UP LIST OF WIRING, AND DISPLAY THE SAME
(93_2)
(93_3)
DRAG LISTED ITEMS
RECOGNIZE DRAGGED WIRING
(93_4)
(93_5)
DROP LISTED ITEMS
RECOGNIZE DROP-DESTINATION-WIRING
(93_6)
ALTER SEQUENCE OF LIST OF WIRING DATA
(93_7)
END

Fig. 94

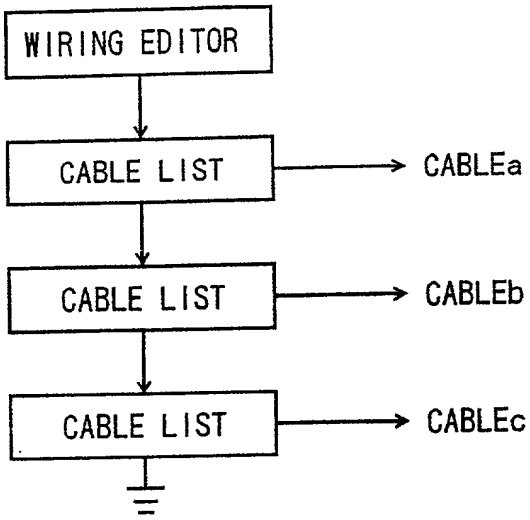


Fig. 95

OBJECT B : BUS1	OBJECT C : BUS2
OBJECT B : BUS1	OBJECT D : BUS2
OBJECT B : BUS1	OBJECT E : BUS1

Fig. 96

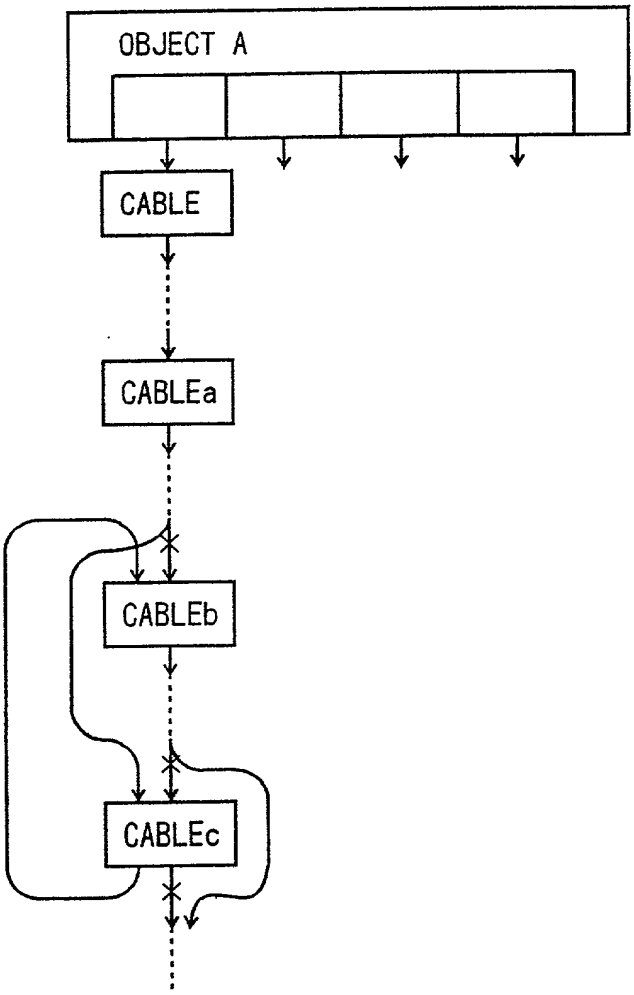


Fig.97

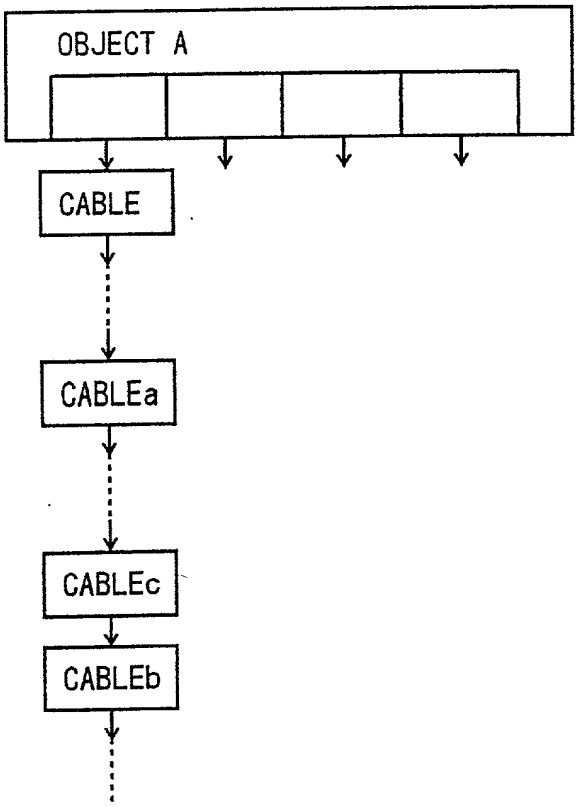

OBJECT A
CABLE
CABLEa
CABLEc
CABLEb

Fig.98

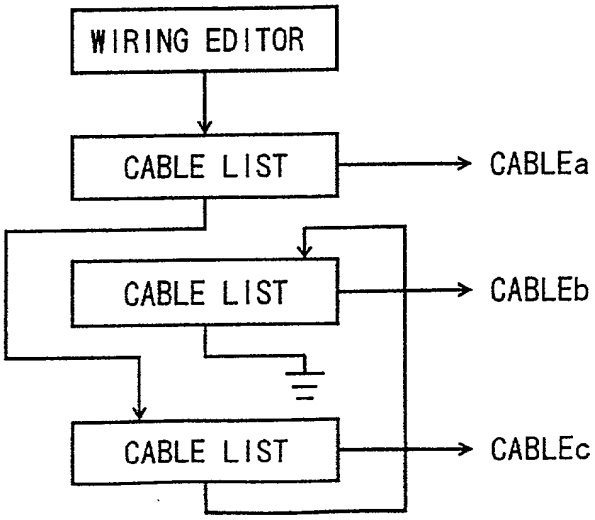

WIRING EDITOR
CABLE LIST
CABLEa
CABLE LIST
CABLEb
CABLE LIST
CABLEc

Fig.99

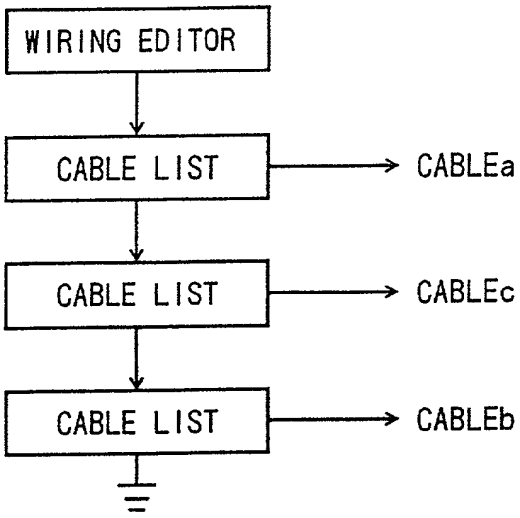


Fig.100

OBJECT B : BUS1	OBJECT C : BUS2
OBJECT B : BUS1	OBJECT E : BUS1
OBJECT B : BUS1	OBJECT D : BUS2

Fig.101

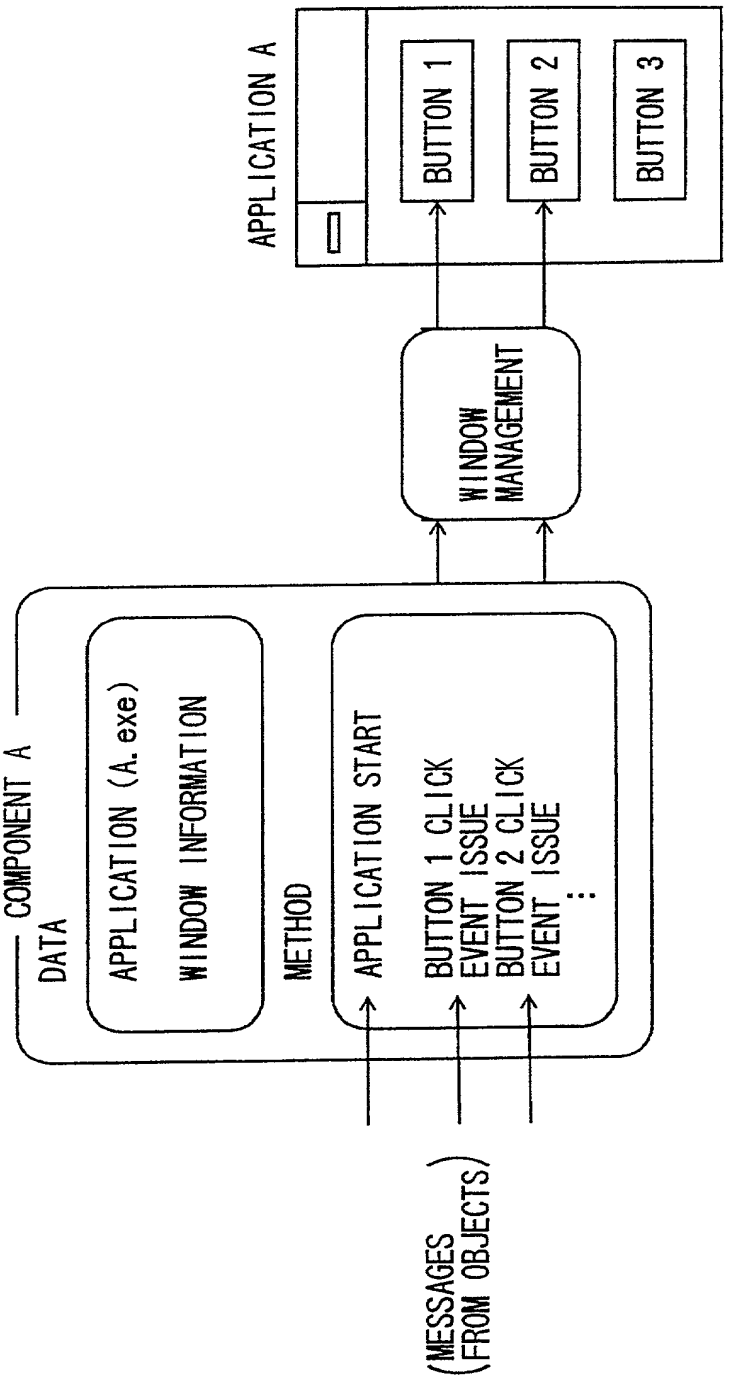

COMPONENT A
DATA
APPLICATION (A.exe)
WINDOW INFORMATION
METHOD
APPLICATION START
BUTTON 1 CLICK
EVENT ISSUE
BUTTON 2 CLICK
EVENT ISSUE
...
(MESSAGES FROM OBJECTS)
WINDOW MANAGEMENT
APPLICATION A
BUTTON 1
BUTTON 2
BUTTON 3

Fig. 102

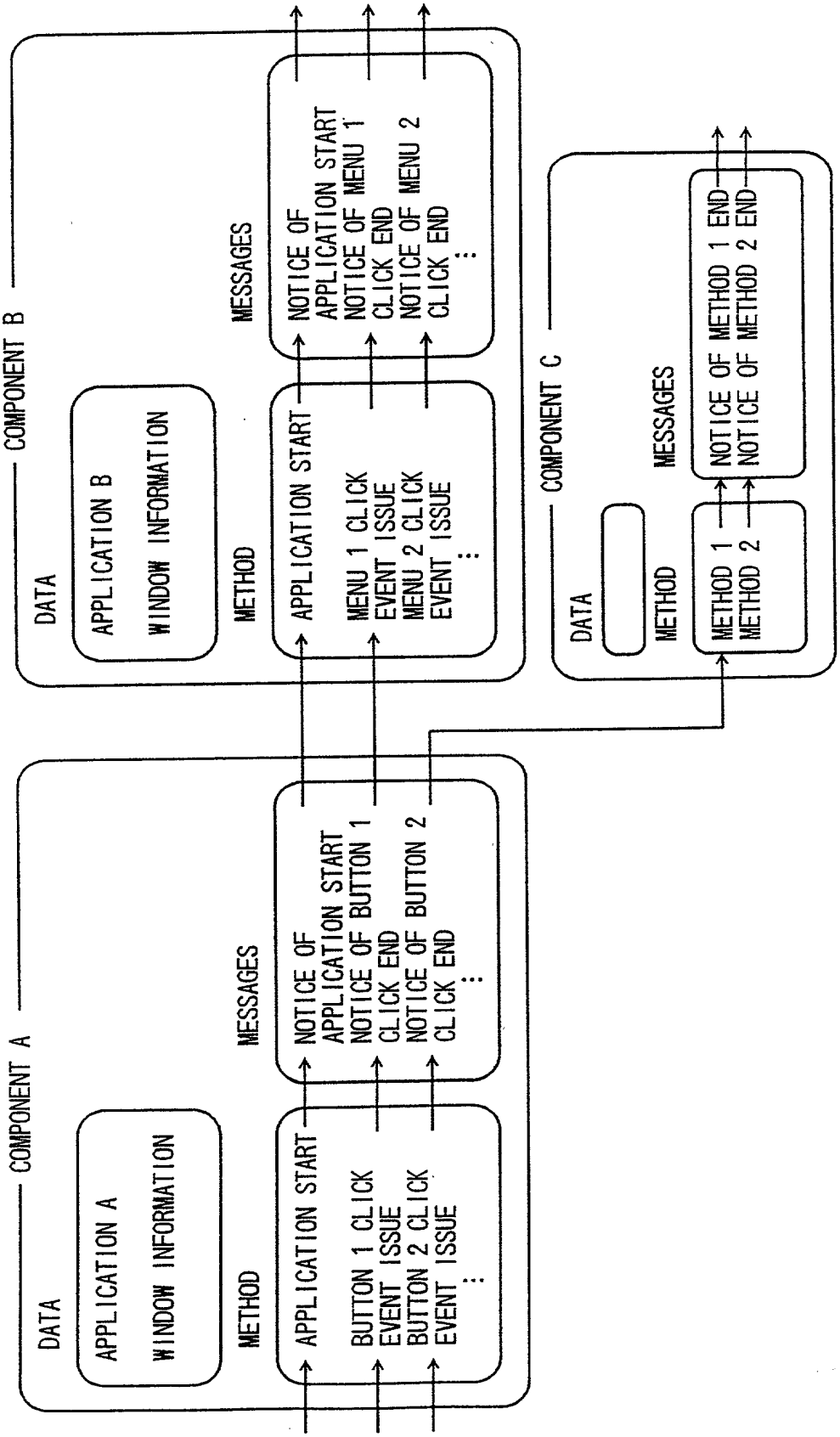

COMPONENT A
DATA
APPLICATION A
WINDOW INFORMATION
METHOD
APPLICATION START
BUTTON 1 CLICK EVENT ISSUE
BUTTON 2 CLICK EVENT ISSUE
MESSAGES
NOTICE OF APPLICATION START
NOTICE OF BUTTON 1 CLICK END
NOTICE OF BUTTON 2 CLICK END
COMPONENT B
DATA
APPLICATION B
WINDOW INFORMATION
METHOD
APPLICATION START
MENU 1 CLICK EVENT ISSUE
MENU 2 CLICK EVENT ISSUE
MESSAGES
NOTICE OF APPLICATION START
NOTICE OF MENU 1 CLICK END
NOTICE OF MENU 2 CLICK END
COMPONENT C
DATA
METHOD
METHOD 1
METHOD 2
MESSAGES
NOTICE OF METHOD 1 END
NOTICE OF METHOD 2 END

Fig. 103

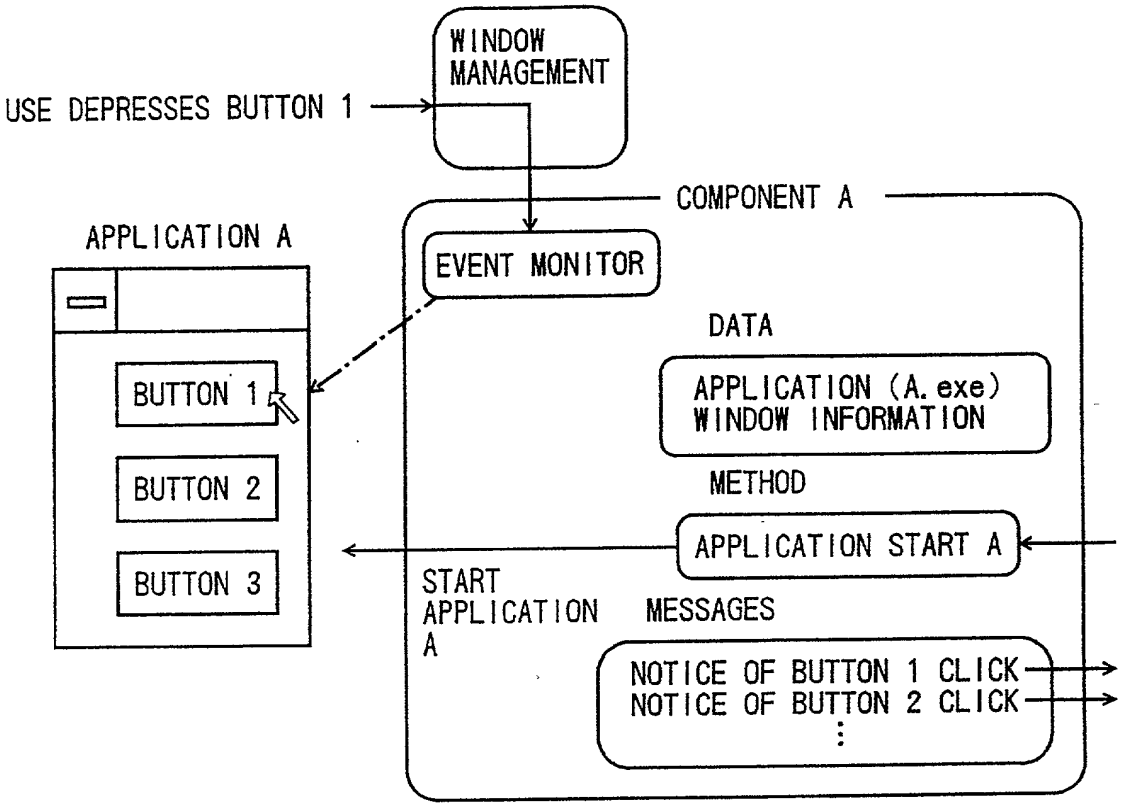

WINDOW MANAGEMENT
USE DEPRESSES BUTTON 1
COMPONENT A
APPLICATION A
EVENT MONITOR
DATA
APPLICATION (A.exe) WINDOW INFORMATION
BUTTON 1
BUTTON 2
BUTTON 3
METHOD
APPLICATION START A
START APPLICATION A
MESSAGES
NOTICE OF BUTTON 1 CLICK
NOTICE OF BUTTON 2 CLICK

Fig. 104

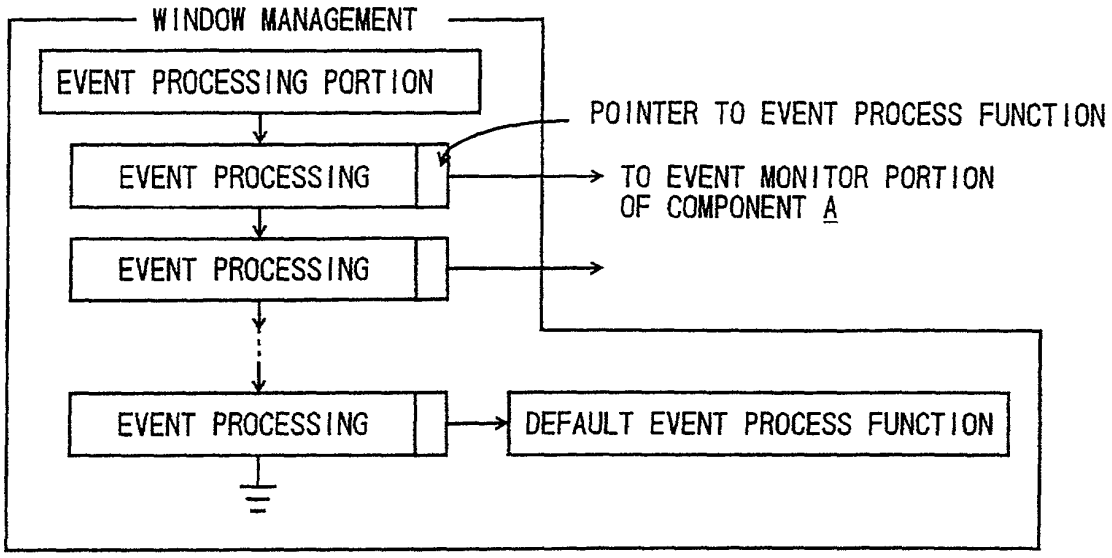


Fig. 105

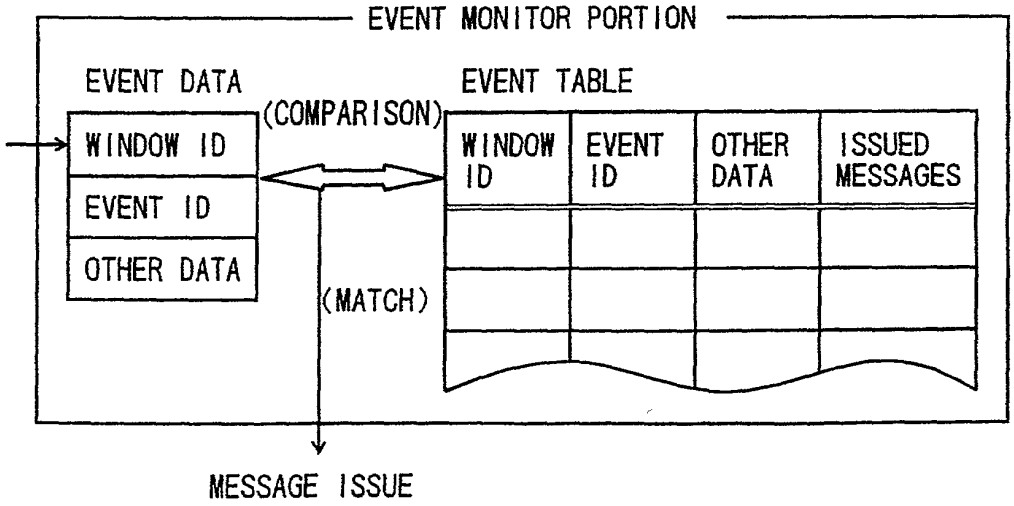


Fig.106

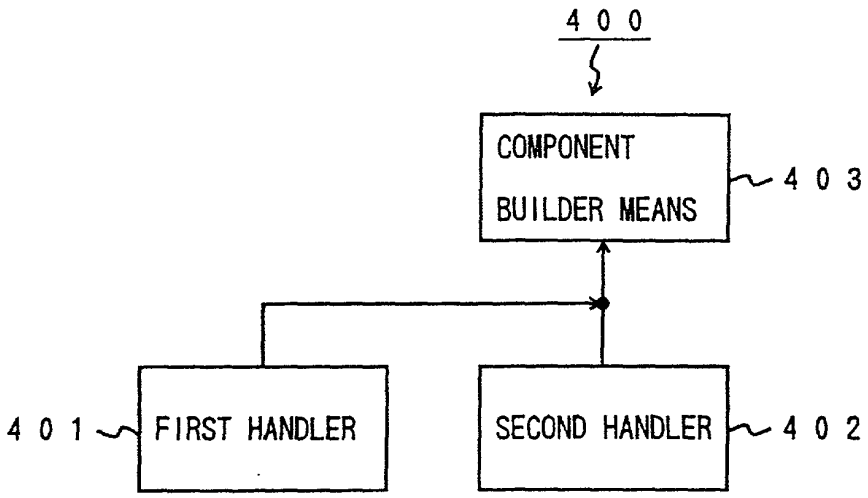


Fig.107

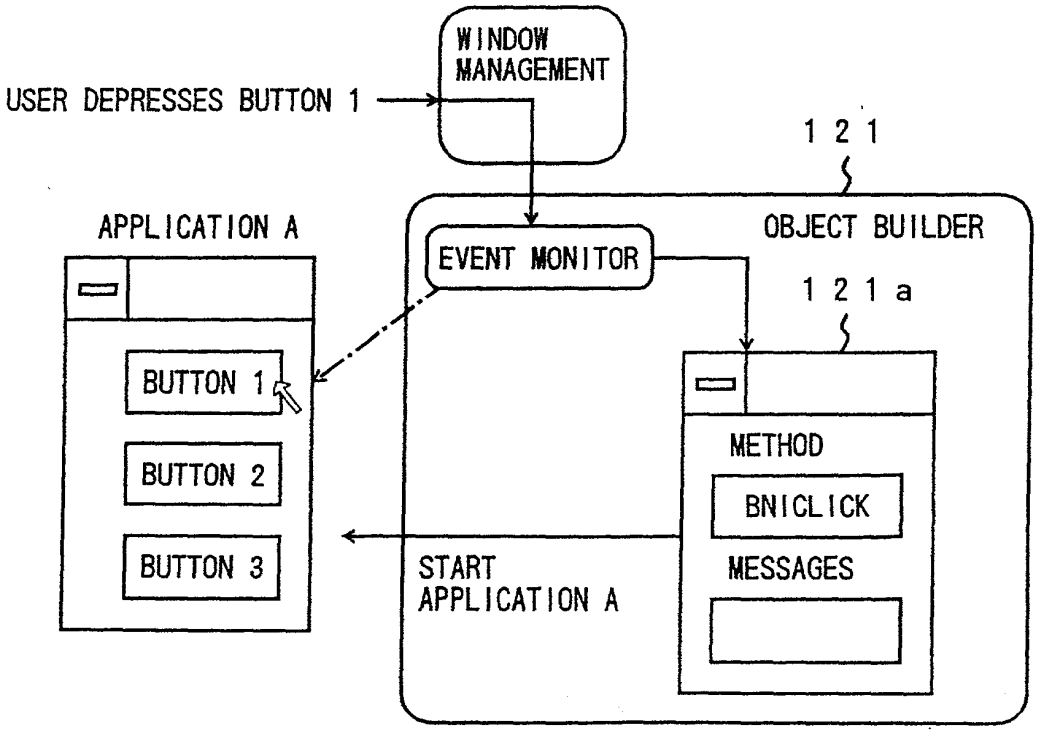


Fig.108

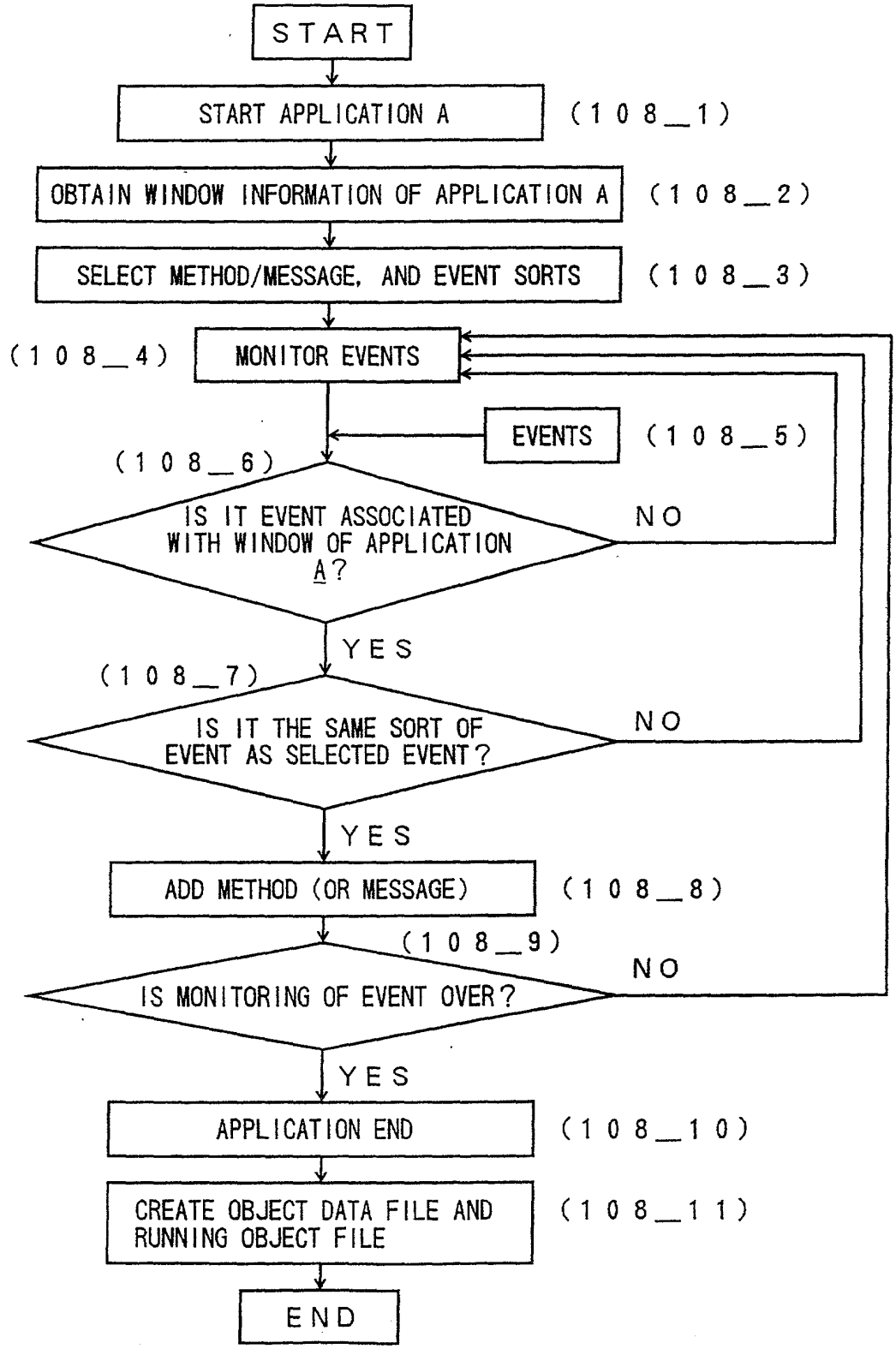

START
START APPLICATION A
(108_1)
OBTAIN WINDOW INFORMATION OF APPLICATION A
(108_2)
SELECT METHOD/MESSAGE, AND EVENT SORTS
(108_3)
(108_4)
MONITOR EVENTS
EVENTS
(108_5)
(108_6)
IS IT EVENT ASSOCIATED WITH WINDOW OF APPLICATION A?
NO
YES
(108_7)
IS IT THE SAME SORT OF EVENT AS SELECTED EVENT?
NO
YES
ADD METHOD (OR MESSAGE)
(108_8)
(108_9)
IS MONITORING OF EVENT OVER?
NO
YES
APPLICATION END
(108_10)
CREATE OBJECT DATA FILE AND RUNNING OBJECT FILE
(108_11)
END

Fig.109

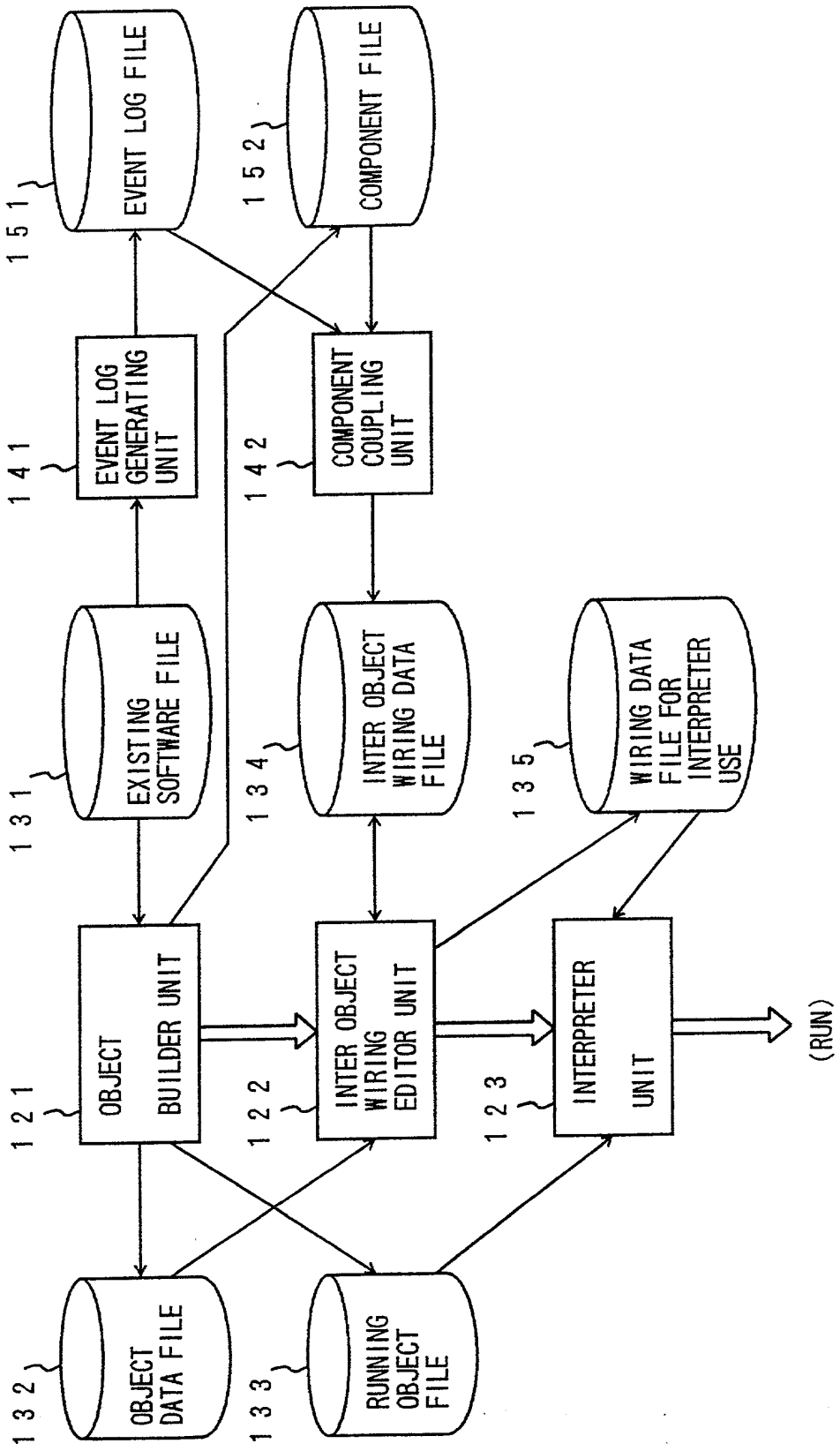

120'
151
EVENT LOG FILE
152
COMPONENT FILE
141
EVENT LOG GENERATING UNIT
142
COMPONENT COUPLING UNIT
131
EXISTING SOFTWARE FILE
134
INTER OBJECT WIRING DATA FILE
135
WIRING DATA FILE FOR INTERPRETER USE
121
OBJECT BUILDER UNIT
122
INTER OBJECT WIRING EDITOR UNIT
123
INTERPRETER UNIT
(RUN)
132
OBJECT DATA FILE
133
RUNNING OBJECT FILE

Fig.110

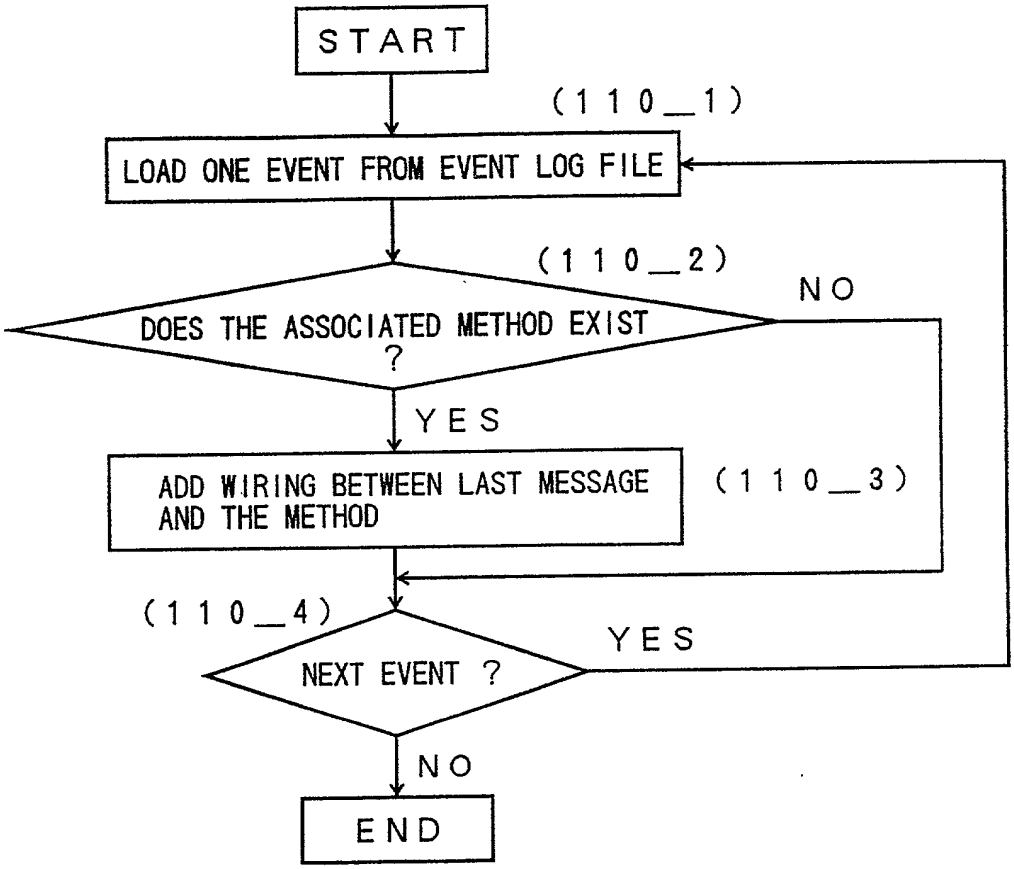

START
(110_1)
LOAD ONE EVENT FROM EVENT LOG FILE
(110_2)
DOES THE ASSOCIATED METHOD EXIST ?
NO
YES
ADD WIRING BETWEEN LAST MESSAGE AND THE METHOD
(110_3)
(110_4)
NEXT EVENT ?
YES
NO
END

Fig.111

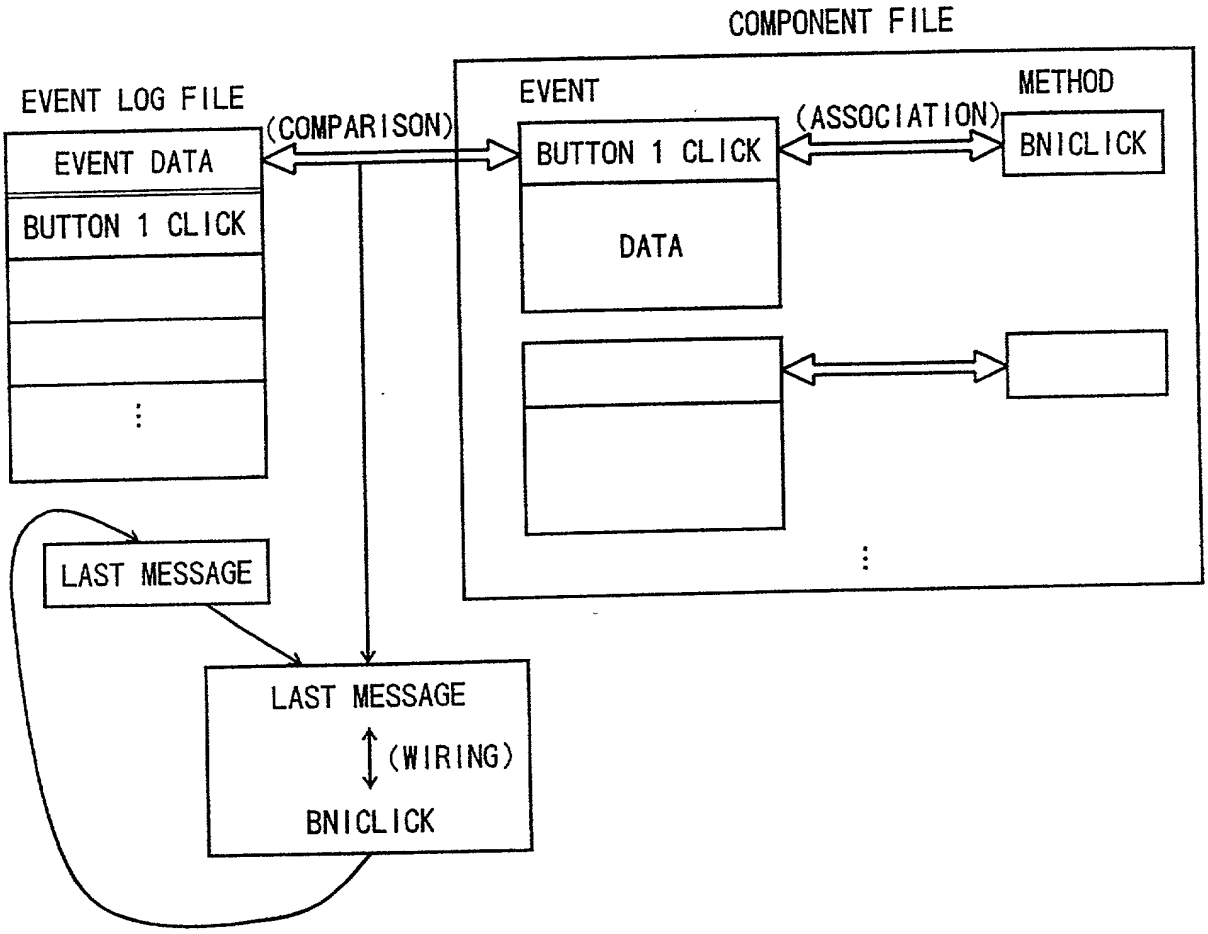

COMPONENT FILE
EVENT LOG FILE
EVENT
METHOD
(COMPARISON)
(ASSOCIATION)
EVENT DATA
BUTTON 1 CLICK
BUTTON 1 CLICK
BNICLICK
DATA
LAST MESSAGE
LAST MESSAGE
(WIRING)
BNICLICK

Fig. 112

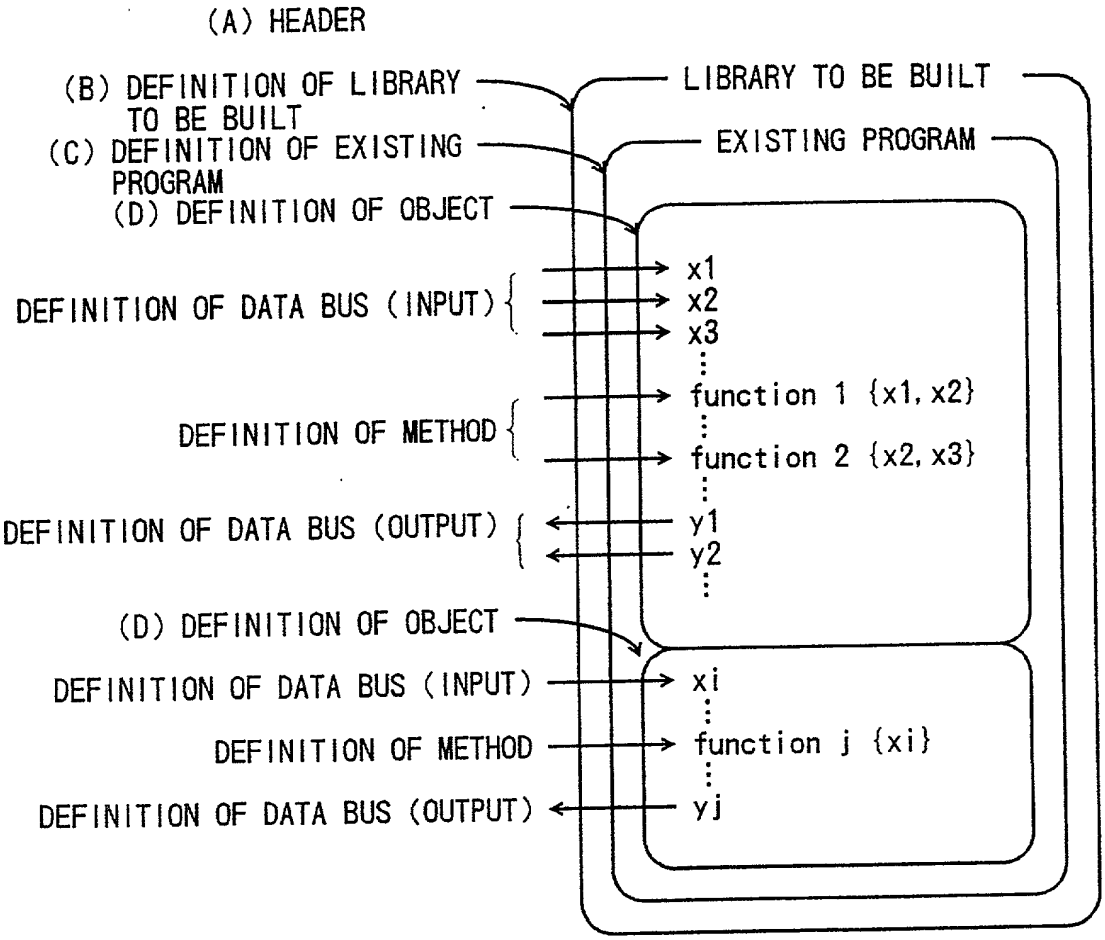
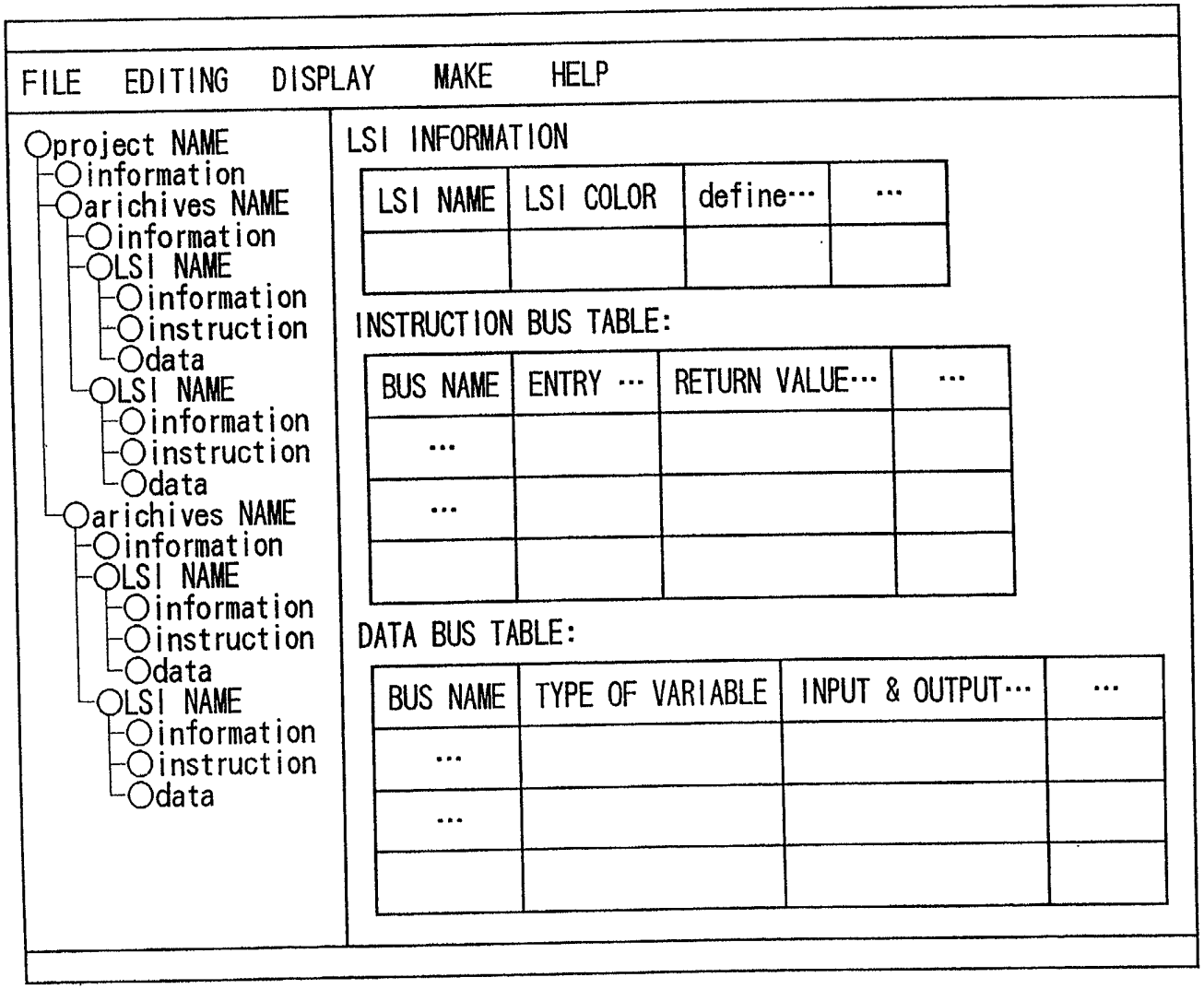

(A) HEADER
(B) DEFINITION OF LIBRARY TO BE BUILT
(C) DEFINITION OF EXISTING PROGRAM
(D) DEFINITION OF OBJECT
DEFINITION OF DATA BUS (INPUT)
DEFINITION OF METHOD
DEFINITION OF DATA BUS (OUTPUT)
(D) DEFINITION OF OBJECT
DEFINITION OF DATA BUS (INPUT)
DEFINITION OF METHOD
DEFINITION OF DATA BUS (OUTPUT)
LIBRARY TO BE BUILT
EXISTING PROGRAM
x1
x2
x3
function 1 {x1, x2}
function 2 {x2, x3}
y1
y2
xi
function j {xi}
yj

Fig. 113

	ITEMS	KEYWORDS	REMARKS
(A)	PROJECT	LSIBuilderProject	
	PROJECT NAME	LSIBuilderProjectName	
	PATH OF COMPILER SYSTEM	MSVCRoot	
	PATH OF FIRSTSIGHT SYSTEM	CoreRoot	
	PATH OF USER AREA	UserRoot	
(B)	DEFINITION OF ARCHIVES	Archives	
	NAME OF ARCHIVES	ArchivesName	
	PATH OF LIB	LibPath	
	PATH OF DLL	DllPath	
(C)	NAME OF LIBRARY TO BE BUILT	LibName	
	COMPILE MODE	Debug	
	DEFINITION OF #define AND typedef	Header	
(D)	DEFINITION OF LSI	LSI	
	NAME OF LSI	LSIName	
	COLOR OF LSI	Color	TREE COLORS OF RGB (0-255)
	DATA BUS	DataBus	
	NAME OF DATA CORRECTION PROCESS	ProcessName	
	NAME OF DATA BUS	Name	
	TYPE OF VARIABLES	VariableType	CODE OF FUNCTION
	DATA CORRECTION PROCESS	Process	
	DIRECT DEVELOPMENT INTO DefineConnector	Inline	
	DISTINCTION BETWEEN INPUT AND OUTPUT	IO	input OR output
	COLOR OF BUS	Color	
	INSTRUCTION	InstBus	
	NAME OF INSTRUCTION BUS	Name	
	FUNCTION NAME OF ENTRY POINT	ProcessName	
	MEANING OF RETURN VALUE	ReturnValue	zero OR nonzero OR NUMERAL
	INSTRUCTION PROCESS	Process	CODE OF FUNCTION
	Cmd ?	Cmd	yes/no
	DIRECT DEVELOPMENT INTO Cmd OR Command	Inline	
	COLOR OF BUS	Colo	
	GLOBAL VARIABLES (GLOBAL VARIABLES INSIDE LSI)	Variables	
	DEFINITION #define AND typedef	Header	
	INITIALIZATION PROCESS	Initialize	CODE OF FUNCTION
	CONSTRUCTOR	Constructor	CODE OF FUNCTION
	DESTRUCTOR	Destructor	CODE OF FUNCTION

Fig.114


FILE EDITING DISPLAY MAKE HELP
project NAME
information
arichives NAME
information
LSI NAME
information
instruction
data
LSI NAME
information
instruction
data
arichives NAME
information
LSI NAME
information
instruction
data
LSI NAME
information
instruction
data
LSI INFORMATION
LSI NAME
LSI COLOR
define…
…
INSTRUCTION BUS TABLE:
BUS NAME
ENTRY …
RETURN VALUE…
…
DATA BUS TABLE:
BUS NAME
TYPE OF VARIABLE
INPUT & OUTPUT…
…